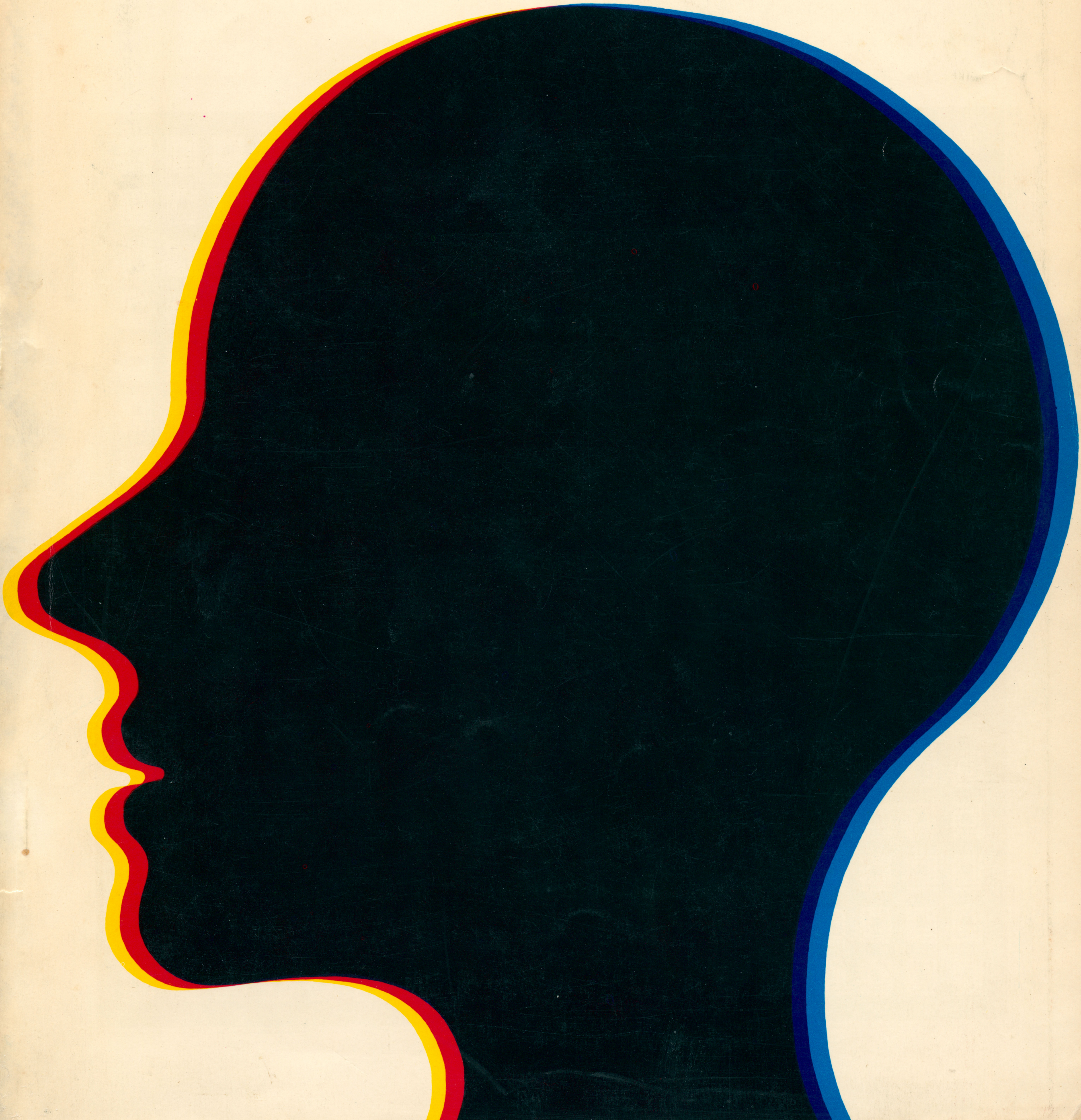
コンピュータ・サイエンス誌 ―― 共立出版

bit

1969 7

昭和44年7月1日発行／毎月1日発行／通巻第5号／昭和44年5月2日第3種郵便物認可／昭和44年1月18日国鉄東局特別扱承認雑誌第3065号



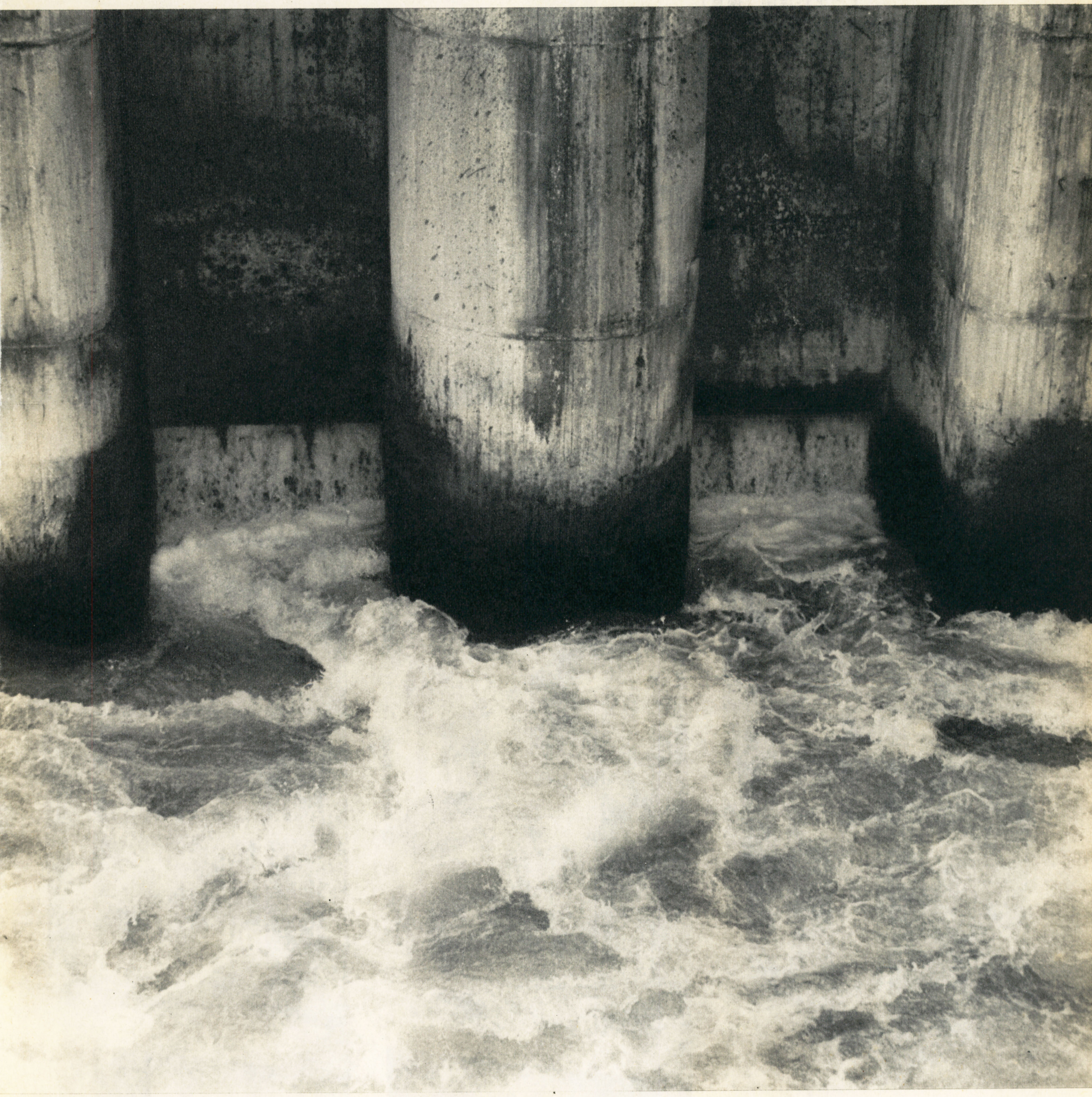
olivetti

《コミュニケーション、その限りない発展のかげで》 人間の英知は伝達の手段を生みました。オリベッティの歴史は、その限りない進歩と発展への挑戦の歴史だったと言えましょう。そしていま、オリベッティは、情報化社会への一翼を担って、未来への扉を開いています。中央から周辺へ、周辺から中央へ、またさらに遠方へ。速く、確実な情報だけが、現代の企業に息吹きを与え、企業の輝かしい未来をつくる――そこにオリベッティの機械があり、オリベッティのシステムがあります。情報革命は、すでにあなたの机の上で始まっています。オリベッティが、インフォメーションに活発な流れを与えます。 日本オリベッティ株式会社 東京都目黒区碑文谷1-26-17

水を治める —— 相模川水系のオンライン制御

堯舜のむかしから治水は〝まつりごと〟のはじめであった。数千年を経た今日でも治水が政治の事始であることに変りはない。ぼう大な人的労力の集積のうえに成しとげられる治水事業も人間が機械と対話することを知った今日、新しく変貌しようとしている。

昭和40年4月、神奈川県津久井湖に完成した城山ダムを指令センターとする相模川水系のオンライン制御システムは(1)データの収集と管理(2)ダムゲート開度の指令(3)貯水池運用の最適化(4)下流取水の確保と利用を総合的に実現した、治水のパノラマである。



相模川——山梨県では桂川とよばれ、神奈川県に入って相模川となり県の中央部を南下して、平塚市東方で馬入川となって相模灘に注ぐ。

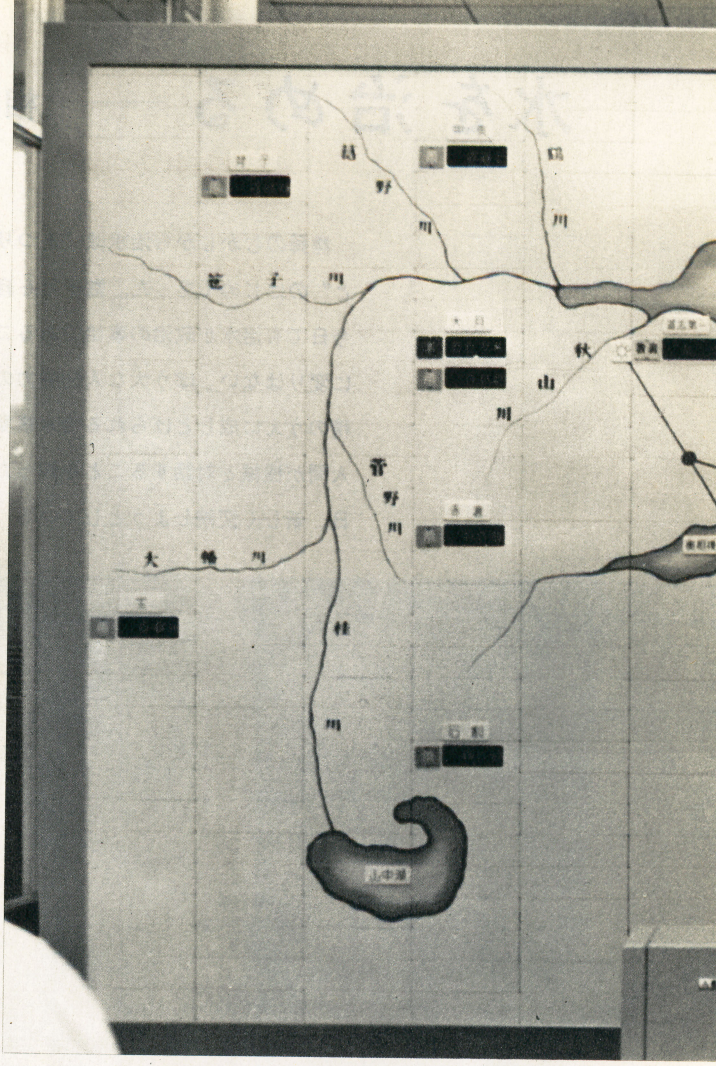
その流域に散在する雨量観測局(7カ所)と水位観測局(14カ所)からの情報に加えて、ダムの水位・ゲート開度・発電電力量など計60量に及ぶデータが5グループの伝送系統を経て城山ダムの電子計算機にインプットされると、計算機は予め記憶している処理手順に従って収集データの計算処理を行ない、結果を必要端末部へフィードバックする。

データ処理例（換算）

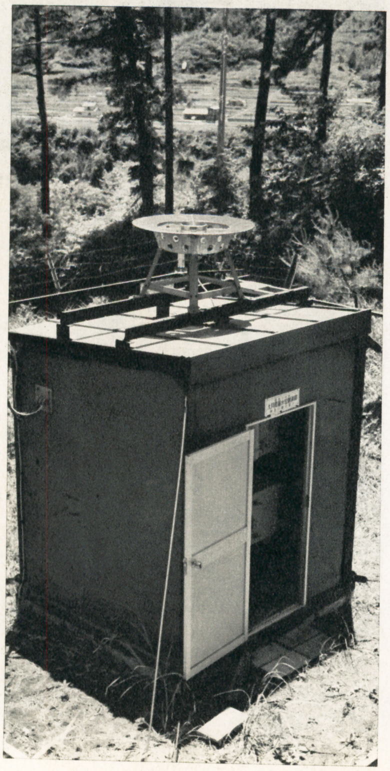
河川水位──→河川流量
雨量──→累計雨量・平均雨量等
ダム水位 ⎫ ⎧ 貯水量・放流量
ゲート開度 ⎬→⎨ 発電所使用水量
発電電力量 ⎭ ⎩ 水道用水取水量

このシステムの最も大きなねらいは
(1)平水時における水道用水の確保
(2)洪水時における最適な流量調節
の2点にある。

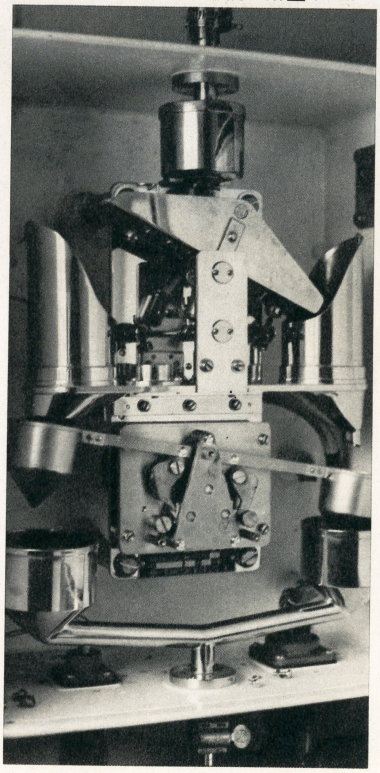
自然現象を対象として、水系諸特性の解明と水系全体を一貫した思想で集中管理するシステムは、通常のプロセスコントロールとはまた別の興味ある多くの問題を含んでいる。



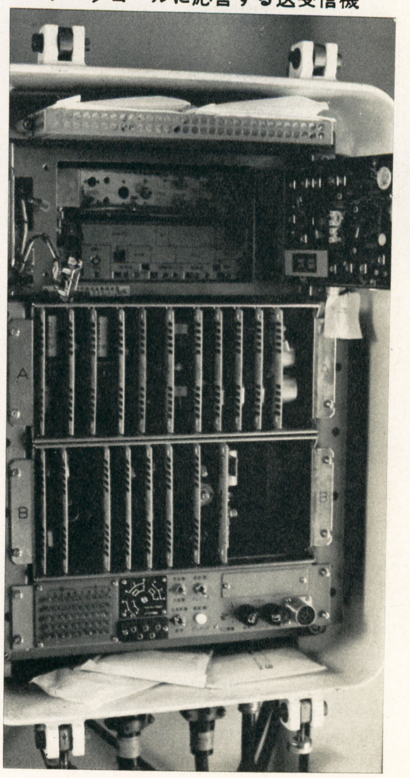
大月雨量観測局

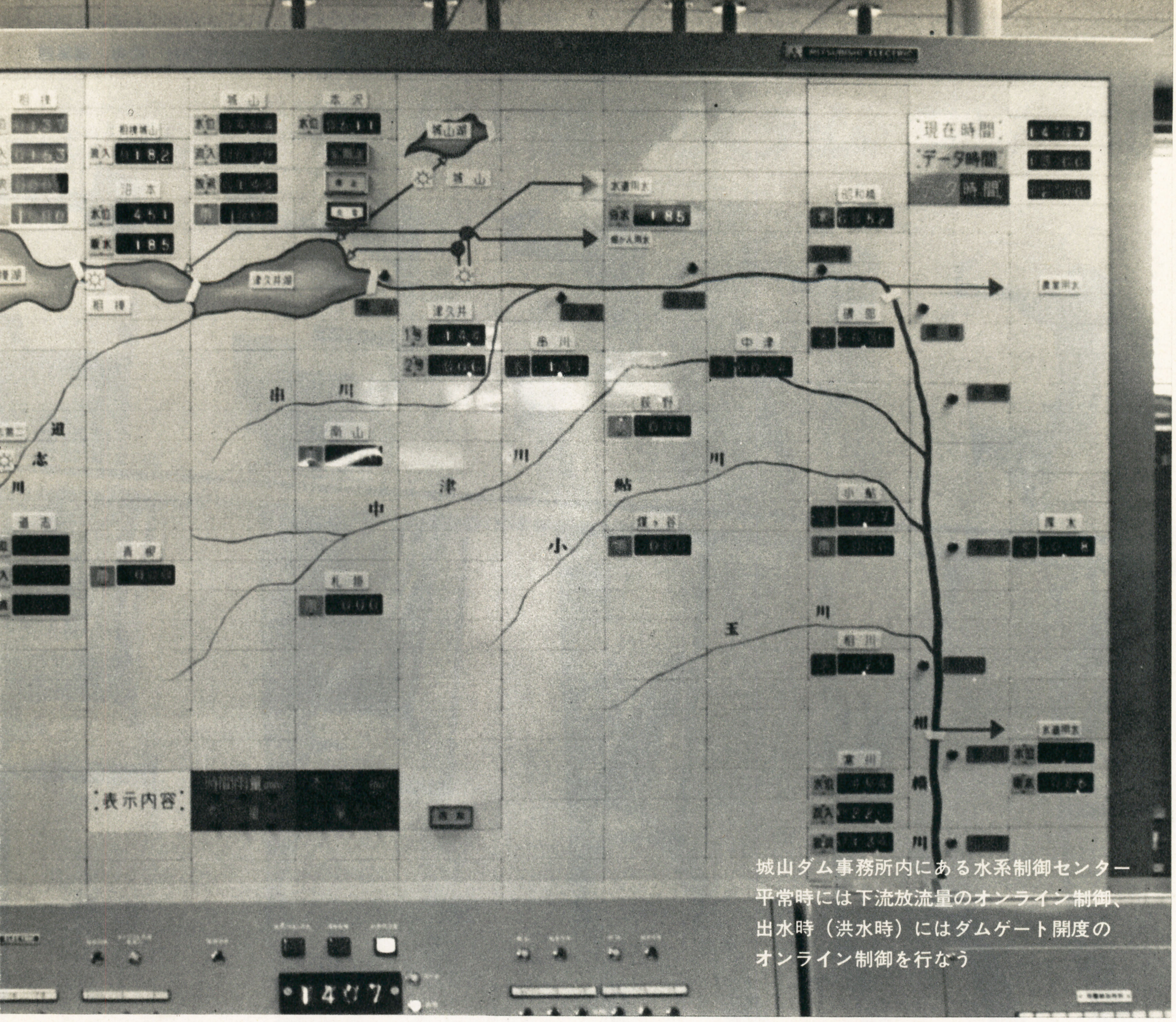


雨量計　円錐部の容積が雨量1㎜



データコールに応答する送受信機

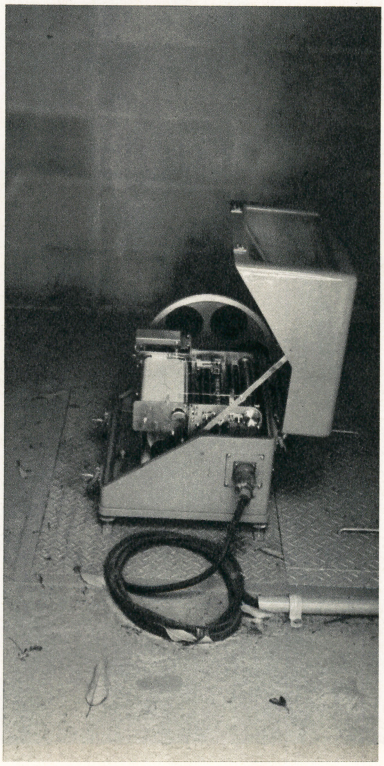




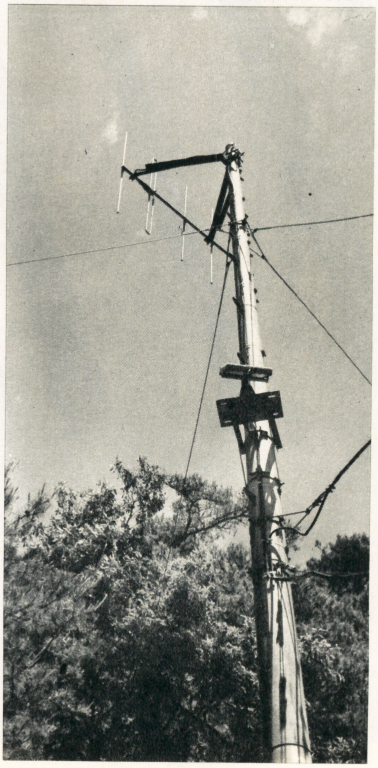
城山ダム事務所内にある水系制御センター
平常時には下流放流量のオンライン制御、
出水時（洪水時）にはダムゲート開度の
オンライン制御を行なう

大月水位観測局

河川水位記録計

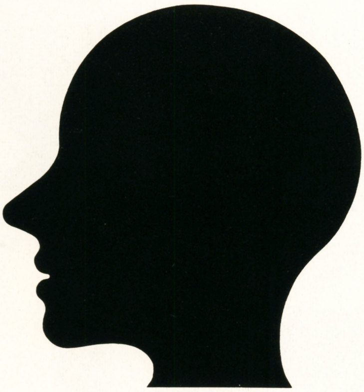


送受信用アンテナ　中央は太陽電池



コンピュータ・サイエンス誌

bit 1969 7



ヒトは
情報を呼吸する
それは文明のいぶき……
機械と対話しながら
未来の序曲をきく

デザイン・構成／沼田　望
カット／原　英夫

情報時代をひらく富士通

流通業の在庫管理に新しい予測システムをご活用ください

FORECAST
《フォーカスト》

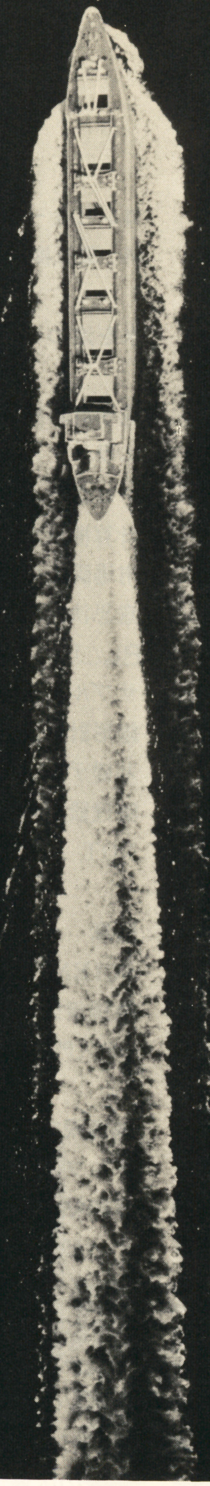
卸売業・小売業などの流通業では、商品の多様化、短い寿命、競争の激化などによって、商品の仕入量、在庫、見切時点を適確に見通すことが非常に重要です。その基礎となる短期の需要予測をするのが、新しく開発されたプログラム＜ＦＯＲＥＣＡＳＴ＞です。予測には、企業内の比較的短期間のデータが使用され、27通りもの計算を自動的に行なって予測をします。●最適予測式を自動決定 ●予測結果が目的別で見やすい ●コボル形式のデータテープがフォートラン形式のデータテープに自動変換されるので、使いやすい。

純国産電子計算機《ファコム》
FACOM

通信と電子の
富士通
Communications and Electronics

富士通株式会社

情報の海のパイロットはユニバック総合研究所がつとめます



コンピューターは　いま情報の海を渡ろうとしている人間が持っている羅針盤ともいうべきもの――その利用技術の開発や普及の必要性が　今日ほど叫ばれているときはありません。
ユニバック総合研究所はコンピューターの　より高度な利用や可能性の追求を目指す専門研究所です

研究開発部門には　情報システム設計に必要な技術の理論的研究を行い　ＭＩＳなどに適用できる汎用システムの設計評価を担当する情報システム研究室をはじめ　デシジョンコントロール技法や行動科学一般の理論的研究を行う経営科学研究室ＯＳ一般論の確立やタイムシェアリングシステムの応用を考えるオペレーティングシステム研究室また　ハードウェアの面からコンピューターの応用を追求していく機器技術研究室など８つの研究室があります

教育部には　プログラム学習やＣＡｉ等新しい教育技術や教育用対話形言語の研究開発を行う教育技術研究室をはじめ各種のセミナーを設け　人材の育成に当るとともにトップから学生まで　コンピューターへの理解を深めるため活動する教育担当など４つの部門があります

４月に発足以来すでに２ヵ月　関係各方面から寄せられる期待の大きさに船足は快調そのもの
企業や社会の安全航海のため　ダイナミックに情報の大海原を先進しています

コンピューターで世界をひらく――

日本ユニバック

本社　東京都千代田区大手町１－１　TEL211 6411
営業所　大阪・名古屋・札幌・仙台・広島・福岡

日本ユニバックは時代の最先端をゆく電子計算機要員を広く求めています。経験は問いません。詳細は本社人事課までお問合せください。応募の秘密は厳守します。

コンピュータ・サイエンスの教育

山内二郎

よくいわれているように，戦後のこの四半世紀の目立つ特長は，いわゆる技術革新のすさまじい発展にある。そのなかでも，われわれに身近かで，その影響の大きく広がったものに電子計算機がある。5年も経つと古くなってしまうといわれるように，その進歩も速い。

そもそもは，戦争中にアメリカで，大量の計算をすばやくこなしてしまうという目的で，研究されはじめ，電子管を使って実現されたので，電子計算機という名前がつき，卓上計算機と区別して，コンピュータといい始めた。生まれた目的がそうであったから，はじめは科学技術の計算に対して，その使い方の研究が行なわれていたのは当然である。しかし電子計算機を使っての処理方法が工夫され，その処理速度が高まり，計算機構が大型化するにつれて，ほかの用途にも使えるように急速に発達していった。したがって，その利用の分野の種類，範囲がさらに拡大してゆき，その周辺機器も開発され，ますますその利用が，いわゆる増殖的に広がってきている。電子計算機機構がなかったら，とてもできなかったようなことも，これでみごとに短時間に処理されて，新しい分野の発達をみせる例も，ここにいちいちあげることもあるまい。

そこでこのように広い範囲にわたるコンピュータ・サイエンスといわれる学問分野が開けてきたのである。けたちがいの財力をもつアメリカはもとよりであるけれども，一昨年秋カナダのトロント大学を訪れたときのことは，カナダですらコンピュータ・サイエンスの教育，研究にひじょうな努力がはらわれているという事実を，痛切に思い知らされたことであった。一般的にいって，カナダの研究所の研究には，わが国を含めて外国からの研究者が大勢参加しているのが，過去の例であった。それほど人口が少ないとみてよい国である。そのような国の大学で，立派な計算センターがあるうえに，これとは独立したコンピュータ・サイエンスの学科があって，しかもその教授の数は24人とのことであった。

わが国の大学には，コンピュータ・サイエンスの学科をもっているのは一つもない現状であり，今日のわが国の大学の制度のあり方では，コンピュータ・サイエンスのように，現存の多くの学部にまたがる学問分野の専門学科を開設することは，ほとんど不可能といってよく，せいぜいその中核となるような部分に限って，ある学部に小規模な学科をつくる努力をするぐらいで始めるより手段がないように思われる。もう一つ考えられることは，いろいろな学部の卒業生が，大学院のなかに組織された数研究科の横断式の特殊な専攻を設けたところで教育を受け，研究を進めていくという形態を実現することであろう。この場合にも現在の大学院の制度のあり方を改めなければならないことがある。将来のことを考えて，いろいろな機会に，この主張を訴えてはいるものの，容易なことでないことを知らされている。

このような事態にあるわが国でも，最近は，本誌に述べられているような MIS とか，あるいは昨年あたりの使節団が報告している PPBS とか，その重要性が一般に報ぜられ，にわかに人々の関心を高め，NHK の教育テレビの講座のテキストが70万部にもおよぶときいたことを考えあわせると，いずれは良い方向へ進んでゆくことではあろうが，情報処理学会の教育調査研究委員会での地道な調査研究が，どうしたら高校，大学の教養課程，専門課程などで，電子計算機活用を理解する人口をふやせるかに焦点を合わせていることも役立つと信じている。

連載　電子計算機とは⑤

計算の実行

小野　勝章・伊理　正夫

プログラムのエラーチェック(虫とり)は根気のいる仕事である。だが，根負けしては，計算機は思う結果を出してはくれない……

エラーを直してもういちどかける

一夫　やあ，次郎，どうだい。このあいだのエラーは訂正してきたね。

次郎　うん。簡単に直ったよ。

一夫　では，三郎君の計算センターへ行こう。

三郎　やあ，いらっしゃい。プログラムの訂正なら，枚数が少ないから，自分でもできるでしょう。次郎さんやってごらんなさい。

一夫　そのまえに，このあいだのエラーメッセージが出たところを，どう直したか見せてごらん。

次郎　まずDIMENSION文にFQAVE(2)，FUTIL(2)というのをつけ加えることだ。

一夫　うん。そうすれば，DIMENSION文がコンマで終わっているというエラーと，配列を宣言なしで使っているというエラーが両方とも解決することになるね。

次郎　次に，READ文に継続行を示すマークをつける。

一夫　それから。

次郎　ハコAのなかの文番号105のIF文を，104に改ためる。

一夫　そう。

次郎　ハコBの文番号105のIF文のコンマをとる(カッコの次のコンマ)。それでおしまいだ(図1)。

一夫　このあいだのエラーメッセージに関するかぎり，これでよさそうだな。それじゃカードの訂正ができたら，三郎君に頼んで計算機にかけてもらいたまえ。

………

三郎　こんどは機械室にはいる必要はないでしょう。結果が出たらもってきてあげますから，控え室で待っていてください*。

テスト計算のチェック

三郎　こんどはプログラムがぶじに通って，計算にかかっているようですよ。

次郎　どうもありがとう。早く結果が見たいな。

一夫　まあそうあわてるな。プログラムのリストはきれいにできたね(図2)。計算の結果はごらんのとおりだ(図3)。

次郎　なんだかあまりかんばしくないみたいだなあ。0と1ばかり並んでいるぜ。

一夫　さあ，ぼんやり眺めてないで，合っているかどうか，チェックしようじゃないか。

次郎　でも，ちゃんと行ってるかどうかはわからないよ。これは乱数で計算したんだろ。

一夫　わかるところから片付けるのさ。あとは兄さんにまかしとけよ。

次郎　それではWRITE文のFORMATがうまくいっているかどうか調べよう。はじめに実行する文は，6番目(図2の左側の番号参照)のREAD文だね。はたしてこのとおり読んだかどうかはわからないが，IPROBだけはあとで書き出されるから，ちゃんと読んでくれたかどうかわかるね。

一夫　読み込んだデータは，かならずWRITE文でふたたび印刷させておくと，チェックのために便利だよ。またアウトプットをあとから見るときに，この計算はどんなパラメータで計算した結果であるかをすぐ確認できるからいいんだ。僕たちのプログラムでは，あまり長くならないように，省略したけどね。

次郎　ともかくこのREAD文は，うまくいってくれたものとして次へ行こう。

一夫　ちょっと。僕らの作ったデータをもういちど復習しといてくれよ。

次郎　ああ。IPROBが9999，IMLとIMU(客到着時間間隔の下限と上限)がそれぞれ1と3，INL(1)とINU(1)(レジAのサービス時間の下限，上限)が1と5，

Katsuaki Ono　電力中央研究所
Masao Iri　東京大学工学部

*三郎君の計算センターで，急ぎの計算はQTA(quick turn around)と指定すれば，十数分でアウトプットがもどってくる。QTAでない仕事は，2～3時間かかる。ほかの計算センターでは，アウトプットがもどってくるまでに半日ないし1日，長いときは数日かかる場合もある。

INL(2)とINU(2)（レジBのサービス時間の下限，上限）も１と５だ。

一夫　それからITINT（中間アウトプットの時間間隔）は１で，毎時間単位アウトプットすることにし，ITMAX（終了時間）は50だったね。

次郎　次に実行する文は７番目のWRITE文で，対応するFORMAT文の番号は96だ（４番目)。これはPROBLEMという文字を印刷させるだけだ。

一夫　これはうまく出ているね。次の実行文は

次郎　次は８番目のWRITE文で，IPROBを印刷させている。これもちゃんと出ている。

一夫　次の実行文は。

次郎　次はもうなにをやっているかわからないよ。ご破算のハコ，スタートのハコ，到着のハコ，…という順で進んでいるはずだが。

一夫　そこでなにをやってきたか手がかりになるアウトプットは。

次郎　ずーっと後のWRITE文まで出てこないね。これは，57番目の文だ。

```
0001              DIMENSION  IREG(2),IQ(2),IEND(2),
                $      INU(2),INL(2),IRSUM(2),IQSUM(2),

           01)   IEY013I SYNTAX

0055              FQAVE(J)=FLOAT(IQSUM(J))/FLOAT(ICLOCK)
                              $
           01)   IEY011I UNDIMENSIONED
0056              FUTIL(J)=FLOAT(IRSUM(J))/FLOAT(ICLOCK)
                              $
           01)   IEY011I UNDIMENSIONED
```

⇩

```
                  DIMENSION  IREG(2),IQ(2),IEND(2),
                $      INU(2),INL(2),IRSUM(2),IQSUM(2),
         →      $      FQAVE(2),FUTIL(2)
```

```
             C ... YOMIKOMI NO HAKO
0006               READ (5,88) IPROB,IML,IMU,INL(1),
                $      INU(1),INL(2),INU(2),ITINT,ITMAX,

           01)   IEY013I SYNTAX
0007                   IRAN
                        $
           01)   IEY013I SYNTAX
```

⇩

```
             C ... YOMIKOMI NO HAKO
                   READ (5,88) IPROB,IML,IMU,INL(1),
                $      INU(1),INL(2),INU(2),ITINT,ITMAX,
         ↗      $      IRAN
```

```
0042          105 IF(ICLOCK-IEND(2)),16,15,16
                $                $
         01)   IEY006I DUPLICATELABEL          02)   IEY013I SYNTAX
```

⇩

```
   ↘  12 IF(IREG(1)) 14,104,14
     104 IF(IQ(1)) 14,14,13
      13 IQ(1)=IQ(1)-1
         IREG(1)=1
         IEND(1)=ICLOCK+IRANSU(INL(1),INU(1),IRAN)
   C ... HAKO B
      14 IF(IREG(2)-1) ↙ 16,105,16
     105 IF(ICLOCK-IEND(2)) 16,15,16
```

図１　エラーをどう直したか　→印は訂正個所

一夫　まずその文のFORMATが正しいかどうか調べてごらん。

次郎　ここで印刷するものは，ICLOCK（時計)，IREG(1)（レジAが空きかフサガリか)，IQ(1)（レジAの行列の長さ)，IREG(2)，IQ(2)(同じくレジBについて)，FQAVE(1)，FQAVE(2)（レジA，Bの行列の平均長さ)，FUTIL(1)，FUTIL(2)（レジA，Bの平均稼動率）で，はじめの五つは整数データ，あとの四つは実数データだ。

一夫　FORMATはいいようだね。

次郎　時計も一つずつ進んでいる。ということは，ITINTが１だということだ。

一夫　では時計が50になったとして，その次の実行文は。

次郎　59番目のIF文で，ICLOCKとITMAXが一致するから，つぎに文番号119のWRITE文（60番目）が実行される。これのFORMAT文は番号97で，ENDという字を印刷させるんだ。

一夫　これもできてるね。

乱数のチェック

次郎　ところで，行列がぜんぜんできていないが，こ

```
FORTRAN IV G LEVEL 1, MOD 3                MAIN                DATE = 69126

      C ...CODING FOR QUEUE SIMULATION WITH ARRAY
      C
            DIMENSION  IREG(2),IQ(2),IEND(2),
           $      INU(2),INL(2),IRSUM(2),IQSUM(2),
           $      FQAVE(2),FUTIL(2)
         88 FORMAT(9I6,I12)
         95 FORMAT(1H ,5I6,4F10.3)
         96 FORMAT(1H1,7HPROBLEM)
         97 FORMAT(1H ,3HEND)
      C
      C ... YOMIKOMI NO HAKO
            READ (5,88) IPROB,IML,IMU,INL(1),
           $      INU(1),INL(2),INU(2),ITINT,ITMAX,
           $      IRAN
            WRITE(6,96)
            WRITE(6,95) IPROB
      C ... GOHASAN NO HAKO
            IICLOK=1
            DO 101 J=1,2
            IRSUM(J)=0
            IQSUM(J)=0
        101 CONTINUE
      C ... START NO HAKO
            ICLOCK=1
            IARRIV=1
            DO 102 J=1,2
            IREG(J)=0
            IQ(J)=0
        102 CONTINUE
      C ... TOOCHAKU NO HAKO
          1 IF(ICLOCK-IARRIV) 10,2,10
          2 IF(IREG(1)) 4,3,4
          4 IF(IREG(2)) 6,5,6
          6 IF(IQ(1)-IQ(2)) 8,8,7
          8 IQ(1)=IQ(1)+1
            GO TO 9
          3 IQ(1)=IQ(1)+1
            GO TO 9
          5 IQ(2)=IQ(2)+1
            GO TO 9
          7 IQ(2)=IQ(2)+1
          9 IARRIV=ICLOCK+IRANSU(IML,IMU,IRAN)
      C ... HAKO A
         10 IF(IREG(1)-1)  12,103,12
        103 IF(ICLOCK-IEND(1))12,11,12
         11 IREG(1)=0
         12 IF(IREG(1)) 14,104,14
        104 IF(IQ(1)) 14,14,13
         13 IQ(1)=IQ(1)-1
            IREG(1)=1
            IEND(1)=ICLOCK+IRANSU(INL(1),INU(1),IRAN)
      C ... HAKO B
         14 IF(IREG(2)-1)  16,105,16
        105 IF(ICLOCK-IEND(2)) 16,15,16
         15 IREG(2)=0
         16 IF(IREG(2)) 110,107,110
        107 IF(IQ(2)) 110,110,17
         17 IQ(2)=IQ(2)-1
            IREG(2)=1
            IEND(2)=ICLOCK+IRANSU(INL(2),INU(2),IRAN)
      C ... RUIKEI NO HAKO
        110 DO 109 J=1,2
            IQSUM(J)=IQSUM(J)+IQ(J)
            IRSUM(J)=IRSUM(J)+IREG(J)
        109 CONTINUE
      C ... TOCHUU JIKAN NO HANBETSU
            IF(IICLOK-ITINT) 18,111,18
      C ... TOOKEI TO KAKIDASHI NO HAKO
        111 DO 112 J=1,2
            FQAVE(J)=FLOAT(IQSUM(J))/FLOAT(ICLOCK)
            FUTIL(J)=FLOAT(IRSUM(J))/FLOAT(ICLOCK)
        112 CONTINUE
            WRITE(6,95) ICLOCK,IREG(1),IQ(1),
           $      IREG(2),IQ(2),FQAVE(1),FQAVE(2),
           $      FUTIL(1),FUTIL(2)
            IICLOK=0
            IF(ICLOCK-ITMAX) 18,119,18
        119 WRITE (6,97)
            STOP
         18 ICLOCK=ICLOCK+1
            IICLOK=IICLOK+1
            GO TO 1
            END
```

図 2 プログラムのリスト

れで，はたしてプログラムはいいのかい。

一夫 こういうこともあるだろうと思って，乱数プログラムのアウトプットを，あらかじめとっておいたんだ（図4）。

次郎 用意がいいな。このとおりの乱数が，かならず出てくるの。

一夫 IRANという数を1007347981としただろう。そうするとこのとおりの乱数が出てくるんだ。*

次郎 たとえば IRANSU(IML, IMU, IRAN)，と書くとどうなるの。

一夫 0と1の間の実数型の乱数が呼ばれるたびに順に一つずつとり出される。それが0と1の区間を，（IMU－IML＋1）等分した区間のどこに当たるかによって，IRANSUの値を決めるんだ。たとえば，IMLが1，IMUが3で，一様乱数が0.65だったら，IRANSUは2だ。

次郎 ではそれで乱数がいくつになるかがわかるから，プログラムを追いかけてみよう。

一夫 プログラムで追いかけてもよいが，第2回目のフローチャート（本誌4，5月号，11ページ）をみたほうが追いやすいだろう。

次郎 はじめの客はレジAの行列につき，ハコ9（文番号9の文）で次の客の到着時刻を求める。これは，一様乱数が0.228だから到着時間間隔は1だ。

一夫 次に，はじめの客がレジAにはいり，ハコBでレジAでのサービス時間を求め

* つまり，計算機で発生する乱数は再現性があるので，たいへん便利なこともあるかわりに，実際使用するときにはこの点に気をつけないといけない。

時刻	レジA	行列A	レジB	行列B	平均行列	平均待ち	レジA 稼動率	レジB 稼動率
PROBLEM								
9999								
1	1	0	0	0	0.0	0.0	1.000	0.0
2	0	0	1	0	0.0	0.0	0.500	0.500
3	0	0	1	0	0.0	0.0	0.333	0.667
4	0	0	1	0	0.0	0.0	0.250	0.750
5	1	0	1	0	0.0	0.0	0.400	0.800
6	1	0	0	0	0.0	0.0	0.500	0.667
7	1	0	1	0	0.0	0.0	0.571	0.714
8	0	0	1	0	0.0	0.0	0.500	0.750
9	0	0	1	0	0.0	0.0	0.444	0.778
10	1	0	1	0	0.0	0.0	0.500	0.800
11	1	0	0	0	0.0	0.0	0.545	0.727
12	0	0	1	0	0.0	0.0	0.500	0.750
13	0	0	1	0	0.0	0.0	0.462	0.769
14	0	0	1	0	0.0	0.0	0.429	0.786
15	1	0	1	0	0.0	0.0	0.467	0.800
16	1	0	0	0	0.0	0.0	0.500	0.750
17	1	0	0	0	0.0	0.0	0.529	0.706
18	0	0	1	0	0.0	0.0	0.500	0.722
19	0	0	1	0	0.0	0.0	0.474	0.737
20	0	0	1	0	0.0	0.0	0.450	0.750
21	1	0	1	0	0.0	0.0	0.476	0.762
22	[illegible]	[illegible]	[illegible]	[illegible]	0.0	0.0	0.500	0.727
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0.0	0.522	0.739
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0.542	0.708
[illegible]	0	0	1	0	[illegible]	[illegible]	[illegible]	0.720
44	0	0	1	0	0.0	[illegible]	[illegible]	[illegible]
45	1	0	0	0	0.0	0.0	[illegible]	[illegible]
46	0	0	1	0	0.0	0.0	0.543	[illegible]
47	1	0	0	0	0.0	0.0	0.553	0.638
48	1	0	0	0	0.0	0.0	0.563	0.625
49	1	0	1	0	0.0	0.0	0.571	0.633
50	1	0	1	0	0.0	0.0	0.580	0.640
END								

図3　アウトプット（一夫と次郎がチェックしたあとがある）

る。次の乱数は 0.146 だから，サービス時間は 1 だ。

次郎　次の客は時刻 2 にきて，レジBの行列につく。ふたたびハコ 9 で,乱数が 0.824 だから，時間間隔は 3 で，次の客は時刻 5 にくることになる。

一夫　二人目の客はレジBにはいり，ハコ17でレジBでのサービス時間は，乱数が 0.632 だから 4 だ。

…………

次郎　だいたい合っているね。

一夫　そうだな。行列ができない原因はこんなことが考えられる。レジA，レジBのサービス時間の平均がそれぞれ 3，レジが 2 カ所あるからレジのサービス能力は 1.5 時間単位当たり一人の割合で客をさばく能力がある。それにたいして客の到着時間間隔の平均が 2 だから，サービス能力のほうが上まわっていることになるね。

次郎　お客にとっては，そのほうがありがたいが，モデルのチェックのためには，行列が少しできるほうがおもしろいな。

一夫　じゃあ，パラメータを少し変えて，もういちどやってみよう。客の到着時間間隔はそのままにしておいて，サービス時間の上下限をそれぞれ 6 と 3 にしてみないか。

次郎　問題番号は9988にしとくよ。では三郎さん，すみませんがもういちどお願いします。

439536295	0.22795808	2
313536885	0.14600194	1
1770361951	0.82438898	5
1357888797	0.63231623	4
804009815	0.37439620	2
1192994309	0.55553126	3
2069361167	0.96362138	5
1679218221	0.78194690	4
40993415	0.01908904	1
165382037	0.07701200	1
623351487	0.29027057	2
104186941	0.04851582	1
1457409207	0.67865896	4
1364321829	0.63531184	4
1511699055	0.70393968	4
1086265165	0.50583160	3
1502234087	0.69953215	4
1384501685	0.64470875	4
1229354271	0.57246268	3
1358061405	0.63239658	4
1379147287	0.64221549	4
347298373	0.16172338	1
[illegible]	0.19040084	1

図4　チェック計算用の乱数　左欄は整数型，中欄はこれを 0 から 1 の間の実数型にしたもので，プログラムで，IRANSU が呼ばれるたびに，これが出てくるものと考えてよい。右欄はこれを 1～5 の間の乱数に変換したもの

システムアナリシスの再検討

一夫　こんどは行列が出てきたぜ(図 5)。

次郎　おや，ちょっと変だぞ。レジBが空いているのにレジAのほうに行列ができてるよ。

PROBLEM 9988 / 時計	レジA	行列A	レジB	行列B	行列A平均	行列B平均	レジA稼動率	レジB稼動率
1	1	0	0	0	0.0	0.0	1.000	0.0
2	1	0	1	0	0.0	0.0	1.000	0.500
3	1	0	1	0	0.0	0.0	1.000	0.667
4	0	0	1	0	0.0	0.0	0.750	0.750
5	1	0	1	0	0.0	0.0	0.800	0.800
6	1	0	1	0	0.0	0.0	0.833	0.833
7	1	1	0	0	0.143	0.0	0.857	0.714
8	1	1	0	0	0.250	0.0	0.875	0.625
9	1	1	0	0	0.333	0.0	0.889	0.556
10	1	0	1	0	0.300	0.0	0.900	0.600
11	1	0	1	0	0.273	0.0	0.909	0.636
12	1	0	1	0	0.250	0.0	0.917	0.667
13	1	0	0	0	0.231	0.0	0.923	0.615
14	1	0	1	0	0.214	0.0	0.929	0.643
15					0.200	0.0	0.933	0.667
						0.0	0.875	0.688
24	1	0	1	0	0.12[illegible]			
25	1	1	0	0	0.160	0.0		
26	1	1	0	0	0.192	0.0	0.885	0.692
27	1	1	1	0	0.222	0.0	0.889	0.704
28	1	0	1	1	0.214	0.036	0.893	0.714
29	1	0	1	1	0.207	0.069	0.897	0.724
30	1	0	1	0	0.200	0.067	0.900	0.733
31	1	1	1	0	0.226	0.065	0.903	0.742
32	1	0	1	0	0.219	0.063	0.906	0.750
33	1				0.212	0.061	0.909	0.758
							0.912	0.735
	1	0	1	0	0.262	0.048	[illegible]	
43	1	1	1	0	0.279	0.047	0.930	[illegible]
44	1	0	1	0	0.273	0.045	0.932	0.727
45	1	0	1	0	0.267	0.044	0.933	0.733
46	1	1	0	0	0.283	0.043	0.935	0.717
47	1	0	0	0	0.277	0.043	0.936	0.702
48	1	0	0	0	0.271	0.042	0.938	0.688
49	1	0	1	0	0.265	0.041	0.939	0.694
50	1	1	1	0	0.280	0.040	0.940	0.700

END

図5 パラメータを変えたら次郎がちょっと具合の悪いところを発見した（○印）

```
      IQ(J)
  102 CONTINUE
C ... TOOCHAKU NO HAKO
    1 IF(ICLOCK-IARRIV) 10,2,10
    2 IF(IREG(1)) 4,3,4
    4 IF(IREG(2)) 6,5,6
    6 IF(IQ(1)-IQ(2)) 8,206,7
  206 IF(IEND(1)-ICLOCK) 8,8,207
  207 IF(IEND(2)-ICLOCK) 7,7,8
    8 IQ(1)=IQ(1)+1
      GO TO 9
    3 IQ(1)=IQ(1)+1
      GO TO 9
    5 IQ(2)=IQ(2)+1
      GO TO 9
    7 IQ(2)=IQ(2)+1
    9 IARRIV=ICLOCK+IRANSU(IML,IMU,IRAN)
C ... HAKO A
   10 IF(IREG(1)-1)  12,103,12
  103 IF(ICLOCK-IEND(1))12,11,12
   11 IREG(1)=0
```

図6 到着のハコをこのように訂正した

一夫　もともとこのプログラムでは，いったん行列についたら，となりのレジが途中で空いてもそっちへ移れないようになっているが……。

次郎　ここははじめからふさがってるほうに並んでいるよ（図5の時刻7で）。

一夫　なるほど。これはちょうどレジBが空きになるとき，同時にお客が飛び込んできたんだな。プログラムでは到着した客をどちらのレジに入れるかを決めるときに，レジがその時刻に空きになるかどうかを見なかったから，こんな現象が起きたんだ。

次郎　では，どう直せばいい。

一夫　ハコ6（文番号6の文）で，行列の長さを比べるとき，長さが等しかったらレジAがこの時刻に空くかどうかを調べればよい。

次郎　レジが空くかどうかは，IEND(・) と ICLOCK を比べればいいね。では……と，こんなふうに直してみようか(図6)。

一夫　さっきから調べるのを忘れていたが，行列が出たところで出そろったから，各レジの行列平均と，平均稼動率の計算がうまくいっているかどうかたしかめてごらん。

次郎　えーと。これはみんなうまくいってるよ。

一夫　それから図5ではまだ行列のできかたが少ない

PROBLEM 9977	時計	レジA	列A	レジB	列B	列A平均	列B平均	レジA稼動率	レジB稼動率
1	1	0	0	0	0.0	0.0	1.000	0.0	
2	1	0	1	0	0.0	0.0	1.000	0.500	
3	1	0	1	0	0.0	0.0	1.000	0.667	
4	1	0	1	0	0.0	0.0	1.000	0.750	
5	1	1	1	0	0.200	0.0	1.000	0.800	
6	1	1	1	0	0.333	0.0	1.000	0.833	
7	1	1	1	0	0.429	0.0	1.000	0.857	
8	1	1	1	0	0.500	0.0	1.000	0.875	
9	1	0	1	1	0.444	0.111	1.000	0.889	
10	1	1	1	0	0.500	0.100	1.000	0.900	
11	1	1	1	1	0.545	0.182	1.000	0.909	
12	1	1	1	1	0.583	0.250	1.000	0.917	
13	1	1	1	1	0.615	0.308	1.000	0.923	
14	1	1	1	1	0.643	0.357	1.000	0.929	
15	1	1	1	1	0.667	0.400	1.000	0.933	
16	1	1	1	1	0.688	0.438	1.000	0.938	
17	1	2	1	1	0.765	0.471	1.000	0.941	
18	1	1	1	1	0.778	0.500	1.000	0.944	
19	1	2	1	1	0.842	0.526	1.000	0.947	
20	1	2	1	0	0.900	0.500	1.000	0.950	
21	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1.000	[illegible]	
[illegible]	1	4	1	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	
[illegible]	1	4	1	4	1.791	[illegible]	1.000	0.977	
44	1	3	1	3	1.818	1.409	1.000	0.977	
45	1	4	1	3	1.867	1.444	1.000	0.978	
46	1	4	1	3	1.913	1.478	1.000	0.978	
47	1	4	1	3	1.957	1.511	1.000	0.979	
48	1	4	1	3	2.000	1.542	1.000	0.979	
49	1	3	1	4	2.020	1.592	1.000	0.980	
50	1	3	1	4	2.040	1.640	1.000	0.980	
END									

図7
訂正後の結果

ようだから*, 思い切って客の到着時間間隔を1または2にしてしまおう。また三郎君の手をわずらわすことになるね。

....

次郎　こんどはどうかな(図7)。

一夫　次郎が指摘してくれた現象は起こっていないようだね。これでうまくいっているかどうかは，家へ帰ってからゆっくり追いかけてごらん。

次郎　プログラムを作るという仕事は，けっこうたいへんなことなんだね。

一夫　これでも，わりとスンナリいったほうだよ。充分チェックして安心がいったら，中間アウトプットの時間を長くとって，本番の計算をいろいろやってみるんだね。

一夫, 次郎　三郎さん，どうもお世話様でした。

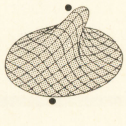
三郎　もうできましたか。早かったですね。ではまた本番のときはいらっしゃい。

*乱数に多少バイアスがあるのか，この部分のサービス時間の平均が，かなり短くなっている。

補足

前号（1969年6月号）12ページ(通巻288ページ)図5 の説明, 上から6行目「ジョブのサブルーチン」を，「ジョブコンのサブルーチン」と訂正。これは，カタログドプロシーデュア（catalogued procedure）とよばれ，一連のジョブコントロール・ステートメントに名前をつけて格納し（カタログする), その名前によってまとめてとり出せるようにしたもので，厳密にはサブルーチンとははたらきが異なる。

次回は「実行プログラムの内容」の予定



アレフ・ゼロ

NHK 教育テレビで4月から始めた日曜日夜7時からの「コンピューター講座」のテキストが全国で70万部も売れたそうです。本誌6月号にもその一部が紹介されているように，各種電子計算機学校が都内だけでも15校もあるそうです。また，計算機セミナーも短期，長期とりまぜていろいろと開かれています。ついでに本誌も○○万部も売れているそうです。いずれもご同慶の至りであります。

もちろん，これらの学校を出たからといって，「講座」を聞いたからといって，本を読んだからといって，何らかの意味で専門家になる（あるいはなれる）わけではありませんから，情報産業に携わるものとして安心してしまうわけにはいかないわけですが，世の中一般の空気が良くなることに異議をとなえる人間はおりますまい。底辺が広くなれば，その山の頂も高くなる可能性があるというものです。もちろん可能性があるからといって高くなるかどうかはわからない点が残念なところです。峨々たる山脈にならずに，坦々とした高原ができ上る可能性もあるわけです。まあ，いずれにせよ，層は狭いより広いほうがいいということはいえるでしょう。

* * *

次の問題は層の頂を高くすることにあるといえるでしょう。一通りの努力は払われているようです。残念ながら一通りというのがそのものズバリのいい方だろうと思われてなりません。たとえば大学における計算機教育はどのくらい充実しているのでしょうか。どんな計算機が使われているのでしょうか。別に大きな計算機があるからといって自慢することもないでしょうが，小さな計算機しかないといって自慢できるものでもありますまい。

現に研究，教育界における計算機密度は，産業界の計算機密度に比べて桁はずれに小さいといってよいのではないでしょうか。大きな計算機が見たかったら銀行に行ってご覧なさい，町の計算機センターに行ってご覧なさい，ということになるでしょう。どうしても金のかかることになるとグッと迫力がなくなるようです。

* * *

この迫力のなくなる話の一つが“日本電子計算機株式会社(JECC)の経営ピンチ”というお粗末です。

ご存知のように，JECC は国産機を保護するため（当然のことですが IBM，GE などに対抗するため），昭和36年に政府が音頭をとり，国産電子計算機メーカー6社が共同出資し，運営しているレンタル会社です。つまり，一種の金融機関になっているわけです。というのは IBM の機械はレンタル制度で使用できる（昔はそれしかできなかった。つまり買取ることはできなかった）ので，金融的にも国産機は不利になってしまう。といってレンタル制度にすれば膨大な資金が必要になるのでとてもまかなえない。そこで JECC がメーカーから国産機を買上げてそれを45カ月の長期分割のレンタルでユーザーに賃貸するという仕掛けをつくったわけです。

そのお陰もあって，現在日本はドイツとともに世界第二位の電子計算機国になっているのだということですが，ついこの間まで5年間で2倍の増加率といわれていたカーブが NHK の70万部ではありませんが，5年間で3倍の増加率になってしまいました。当然 JECC の必要資金量も増加の一途をたどります。たとえば次の表を見てください。

年度	計算機納入実績	JECC 扱い	JECC 支払	未払金累計
	億円			
41	515	268	221	47
42	668	368	330	85
43	1088	666	460	291

*1　41年度＝42年3月末日まで
*2　納入実績＝外国機を含む。買入価格換算
*3　JECC扱い＝納入実績中，JECCが扱ったもの。国産機のほとんどを含む。
参考　IBM社の43年度売上げは68億ドル（24,480億円）

というわけで43年度現在の資本金180億円を上回わる未払金がたまったわけです。さらに44年度には700億円に達するだろうという予想がたてられています。“はたしてこの難局をいかに乗り切るか”という大問題が生じたところです。

“行政ベースの話合いでは事態は改善される期待はできない，政治的に解決せよ”という声があります。電子計算機と政治の関係のきわめて現実的な側面はこんなところにあるわけです。草の根のコンピューター講座テキストと政治家の電子計算機と情報産業と，というお粗末でした。

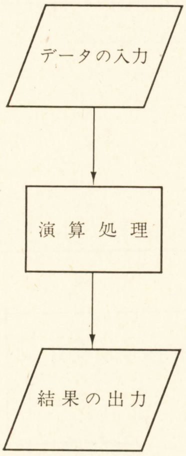
(J. A.)

連載　プログラミング入門⑤

入出力命令

山本　欣子

前回は四則演算のプログラムを勉強した。今回は入出力の命令の使い方について説明しよう



電子計算機で処理するたいていの仕事には，何らかの入出力操作が伴うのがふつうである。標準的な処理の流れは左のようになる。

ところで，入出力のプログラムというのは，かなり面倒である。そのせいか，たいていのプログラミングテキストでは，入出力に関する勉強は，だいぶあとのほうですることになっている。

しかし，勉強している側ではどうも入出力が伴わないと，完全なプログラムにならないような気がして実感がわかない。そこで今回は思いきって入出力の命令の使い方を説明することとした。

かなり約束ごとが多いので退屈するかもしれないがある程度，我慢していただかねばならない。

1.　整数の入力

例題1　1枚のカードにパンチされている3個の数値を読み込んで，その合計を計算し，入力データとともにプリントせよ。ただし数値はすべて5桁ずつの整数とする。

カード上に数値が図1のようにパンチされているとしよう。1枚のカードは80欄あり，それぞれをカラム(column)とよぶことはすでに述べたとおりだが，この例では5桁の数値がそれぞれ5カラムずつパンチされ，その区切りが1カラムずつあいている。

Kinko Yamamoto
日本情報処理開発センター

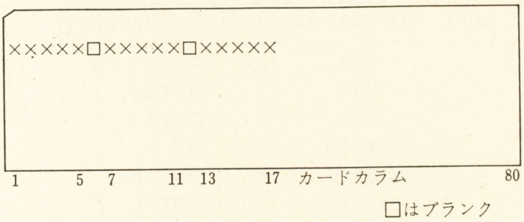


図　1

さて，カードを読み込むには，READ命令を使う。

この命令で，カード1枚分のデータが記憶装置に読み込まれる。このとき80桁分が一度に読み込まれるので，その80桁分のデータのうち，必要なカラムの数値だけを指定した記憶場所に取り出さなければならない。

この例題では，表1のように書けばカード上の三つの数値が，それぞれDATA1，DATA2，DATA3にはいることになる。

INI　DATA 1，1，5

のINIはIN Integerの略で，オペランド部のDATA 1，はDATA 1に読み込めという記憶場所の指定，次の1，は**1カラムから**の意味で最後の5は**5カラム分**ということである。つまりこの一つの命令でカード上の1から5カラムまでの整数値をDATA1に入れよ，ということとなる。同様に次の二つの命令も7カラムから11カラムまでおよび13カラムから17カラムまでの整数をそれぞれDATA 2とDATA 3に入れることになる。

最初のREAD命令でまずカードを読むと記憶装置のどこに読み込まれるかと心配する人がいるかもしれないが，これは記憶装置内の特別な領域で，使用者が使う領域とは別になっているのがふつうである。多くの計算機では，入出力用にとくに，こういった特殊の領域をもっているものが多く，**バッファ**(buffer)とよんでいる。

表1　例題 1 のカード読込み命令

LABEL	OPERATION	IND	OPERAND	説　　明
	READ			カードを1枚読む
	INI		DATA 1，1，5	1～5カラムの整数をDATA 1へ
	INI		DATA 2，7，5	7～11カラムの整数をDATA 2へ
	INI		DATA 3，13，5	13～17カラムの整数をDATA 3へ

表2　例題 1 の合計演算命令

LABEL	OPERATION	IND	OPERAND	説　　明
	LAD		DATA 1	(DATA1)→AR
	ADD		DATA 2	(DATA1)+(DATA2)→AR
	ADD		DATA 3	(DATA1)+(DATA2)+(DATA3)→AR
	STR		TOTAL	(AR)→TOTAL

表3　例題 1 のプリント命令

LABEL	OPERATION	IND	OPERAND	説　　明
	OUTI		DATA 1，5	DATA 1の整数5桁を出力バッファへ
	OUTI		DATA 2，7	DATA 2の整数7桁を出力バッファへ
	OUTI		DATA 3，7	DATA 3の整数7桁を出力バッファへ
	PRINT		*	頁きりかえ後出力バッファ内容をプリント
	OUTH		TOTAL	TOTALの文字を出力バッファへ
	OUTI		TOTAL，7	TOTALの整数7桁を出力バッファへ
	PRINT		1	1行改行後，出力バッファ内容をプリント

さて，ひとたび数値がDATA 1とか，DATA 2にはいってしまえば，あとはしめたもので，これらの合計を計算するのは，いままでの知識で充分できる。すなわち**表2**のプログラムでDATA 1とDATA 2とDATA 3が加え合わされて，合計がTOTALにはいることになる。

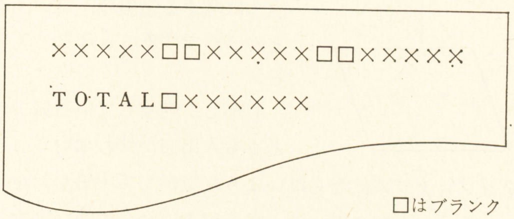
2.　整数の出力

さてこんどは結果のプリントである。電子計算機でふつう使われるプリンタは**ラインプリンタ**（line printer）といって，プリント用紙の1行分を一度にプリントするものである。プリント用紙の1行には，120字ないし160字分，プリントできるくらいの幅があり，その字数はラインプリンタの性能によってそれぞれ決まっている。

いずれにせよ，1行分にプリントすべきものを，ためておいて，全部そろったらPRINT命令でプリントする。

これは入力のカードを読む場合とまったく逆である。この例では入力データと合計とを**図2**のようにプリントすることにしよう。5桁ずつの三つの入力データを1行目にプリントし，次の行に合計を印刷する。この場合，ついでに TOTAL という見出しも プリントさせてやろう。この具体的な命令は**表3**のようになる。

入出力のプログラムは，概してかなりめんどうである。とくにプリントのプログラムは応々にしてだいぶ長くなることが多い。ラインプリンタは1行分一度にプリントするとはいえ，見やすくするためには，数字と数字の間に適当にブランクを入れたり，見出しも印刷したり行と行の間は何行改行するかなどということをプログラムでコントロールしてやらねばならない。



□はブランク

図　2

ラインプリンタ用紙はふつう**図3**のようにほぼ30cmおきくらいに(66行分)にミシン目がはいっている。ミシンからミシンの間を1ページとよび，仕事が変わるたびに，新しいページにページきりかえをやって，プリントを始めるのがふつうである。もちろんページのきりかえをやらなくても，プリントはしてくれるが，まえの仕事の結果と同じページにプリントされたりすると見にくくなるしページをかえれば，そこのミシン目から切り離せて，何かと都合がよい。紙の両側の小さな穴は送り穴で，これで1行ずつ紙送りが行なわれることになる。

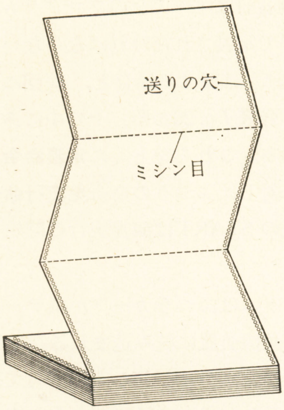


図 3

さて，だいぶ横道にそれたが表3の説明にもどろう。最初の命令 OUTI DATA1, 5 の OUTI は Out Integer の略である。これはDATA 1 の整数を5桁プリントせよということである。同様に次の二つの命令はDATA 2 を 7桁，DATA 3 も 7桁，それぞれプリントせよということになる。

さきほども述べたようにラインプリンタでは1行分をどこかへためておいて PRINT 命令でいっせいにプリントすることになるので，実際のプリントは，4番目の PRINT＊という命令で行なわれる。したがって，そのまえの三つの OUTI 命令では実際のプリントが行なわれるわけではなくて，出力用のバッファに数値が移されるだけである。このバッファはカード読込の場合と同じように記憶装置の特別な領域にある。

PRINT のオペランド部＊ はさきほど述べたページのきりかえを指定する印で，まず，新しいページにきりかえてからプリントせよということである。

次の命令 OUTH TOTAL は TOTAL という字がそのままプリントされる便利な命令である。OUTH のHは character の略で，この命令は，オペランド部に書かれた文字をそのまま印刷する。最後の PRINT 1 は，さきほどと同様，直前の二つのプリントを具体化するものだが，1はプリントする前に1行改行せよ，という指定である。一般に PRINT n とすれば，n 行改行してからプリントするということである。

さて，もう一度，図2を振り返ってみよう。これによると1行目は，5桁ずつの整数の間に，ブランクを二つずつ入れることになっている。同様に2行目も TOTAL と数値の間にブランクを一つ入れることになっている。DATA 1，DATA 2，DATA 3 は，すべて5桁の数値で

海洋時代をささえる
海のサイエンス誌――

月刊 海洋科学

9月創刊号 8月20日発売
毎月20日発売，￥500

表紙／合成紙Q'KOTE 190μ (160kg)カラー
本文／特黄菱 90kg 全頁オフセット2色刷り
仕上り／変型 242×212mm 120頁 上製

創刊にあたり――
本紙は 海洋の基礎科学を主題に 技術と開発の現状 およびあすの姿を論ずるために創刊されます 海洋開発をがっちりとささえ 海洋時代へ正しくみちびく……………
そんな雑誌をつくりたいと念願しています

■月刊《海洋科学》の個性
1. 最高権威者の論文を中心とした 総合的な海洋の基礎科学誌です
2. 毎号ひとつのテーマを特集で追い 徹底的に深く掘りさげています
3. 海洋科学技術の新しい分野を意欲的にとりあげ 理論の体系化を試みています
4. 内外の動きに大幅な頁数をあてて 資料として役立つよう各国別に報道します
5. あすの自然科学誌として 見やすく理解しやすいよう編集してあります

■各号の特集テーマ
創刊号・海水の運動 第2号・海水の物性
第3号・海水の無機 第4号・大陸棚
第5号・深海底 第6号・プランクトン
■読者対象 学生・科学者・技術者・海洋産業従事者・一般社会人

■海の双書《イルカぶっくす》の企画もすでにできており近くシリーズに発売されます
■「海洋図書は海洋出版へ」という信頼を築きたいと一同努力しております

海洋出版株式会社
東京都港区三田4―14―8 ☎(452)1327(代)〒108

あるのに，2番目，3番目の命令は，それぞれ7桁指定している。実際の数値は5桁ずつであるから，7桁と指定すると頭に0が二つついた形になり，それがプリント用紙上では不要な頭の0はサプレスされて表われないので，結果的に二つブランクがあくことになる。同様に，2行目のTOTALの数値は，実際は6桁以内であるが，TOTAL，7と指定してあるので一つないし二つブランクがあくこととなる。ただし，もし数値がマイナスならば，ブランク1個分が－とおきかえられる。

3. 完全なプログラム

さて，これで例題1のプログラムの道具立ては全部そろった。全体をまとめると**表4**のようになる。

カード読込，合計演算，プリント命令と続き，最後にHALT命令がついている。これは実行を終わらせる停止命令である。その後のDATA1 RS1は，DATA1という記憶場所を1語確保せよということである。

RSはReserve Storageの略で，1は1語のことである。一般にRS n で n 語確保されることとなる。

次のDATA2，DATA3，TOTALもすべて同じで，それぞれ1語ずつ確保される。

このように，あるプログラムのオペランド部に出てくる記憶場所の名前はすべてそのプログラムのどこかで，このような形で定義されなければならない。それには，RSで場所だけを確保する場合と，前回に述べたように（本連載6月号**表7** p. 20参照）その中にある数値を入れる場合とがある。これらをとくに**定義命令**とよぶことがあり，それに対し，ふつうの命令を**実行命令**という。

さて，最後のSTARTは演算実行の開始を指定するものである。

このプログラムはふつうカードか紙テープにパンチされ一応全部，記憶装置に読み込まれる。そしてSTARTの指令で，最初の命令READから始まって，書かれた順に各命令を実行し，最後のHALT命令で停止する。

ところで，このDATA1とかDATA2とかいう記憶場所の定義命令を実行命令群の後につける場合と，前につける場合とがある。いずれにしろこの例のように一まとめにしておくほうがよいが，これを後につけるか，前につけるかは習慣のようなもので，とくにこだわる必要はない。

しかしここでは後につけるのを標準としておこう。もしこれを前につけたとすると，演算の実行開始を指定する最後のSTART指令を少し変えねばならない。

STARTだけだと最初の命令から機械的に開始してし

表4 例題1のプログラム

LABEL	OPERATION	IND	OPERAND	説明
	READ			カード読込（表1）
	INI		DATA1，1，5	
	INI		DATA2，7，5	
	INI		DATA3，13，5	
	LAD		DATA1	合計演算（表2）
	ADD		DATA2	
	ADD		DATA3	
	STR		TOTAL	
	OUTI		DATA1，5	プリント（表3）
	OUTI		DATA2，7	
	OUTI		DATA3，7	
	PRINT		*	
	OUTH		TOTAL	
	OUTI		TOTAL，7	
	PRINT		1	
	HALT			停止
DATA1			RS1	
DATA2			RS1	
DATA3			RS1	
TOTAL			RS1	
	START			最初の命令から実行開始

表5　定義命令を前につけたプログラム

LABEL	OPERATION	IND	OPERAND
DATA 1			RS 1
DATA 2			RS 1
DATA 3			RS 1
TOTAL			RS 1
KAISI	READ		
	INI		DATA 1, 1, 5
	INI		DATA 2, 7, 5
	⋮		
	HALT		
	START		KAISI

まうことになるので，これが実行命令でない場合は，たとえば**表5**のように，開始命令に適当なラベル（ここではKAISI）をつけ，それをSTARTのオペランド部に指定する。

さて，それではここでいままでの入出力命令を**表6**（次ページ）にまとめて，今回の終りとしよう。

次回は「実数の演算と入出力」の予定

お詫びと訂正

本連載の5月号で，年月日を与えると，その曜日を計算するプログラムのフローチャートで，説明不足と一部誤りがありましたので訂正致します。

最初の年月日の入力は，いずれも2桁ずつで，たとえば1969年6月1日なら，Y=69，M=06，D=01となります。

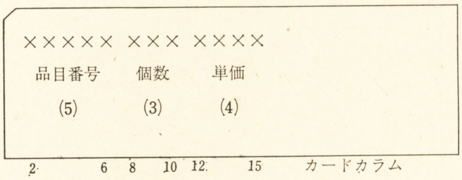
また，下の説明文で，平年と閏年が逆になっています。すなわち，Yが4で割り切れれば**閏年**，割り切れなければ**平年**ということになります。

ご指摘下さいました松下電器のシラガ様にお礼申し上げるとともに，筆者の不注意をお詫び致します。

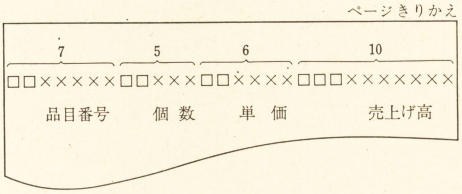
練習問題　（解答は79ページ）

1. ある販売会社の売り上げデータが次のようにカードにパンチされてはる。売り上げ高を計算し，入力データとともに図のように印刷せよ。なお，
　売り上げ高＝単価×個数　で計算する。

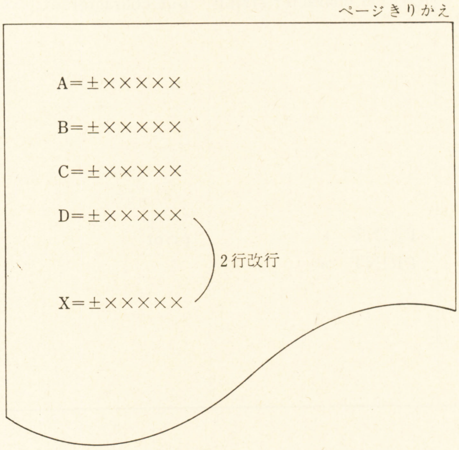
入力データ



印刷形式



2. $a=61527$, $b=74$, $c=-308$, $d=157$ のとき
$x=a-(b^2+cd)$
を計算し a, b, c, x を図のように印刷せよ。



日本コンピュータ協会附属学校
電子計算機セミナー

慶応大学工学部電気工学科
東京工業大学名誉教授　工学博士

学長　尾本　義一

日本コンピュータ協会では、広くわが国における電子計算機教育の普及をはかるため、プログラミングを初めて学ぶ者が自分自身で自由にプログラムが組めるようになる事を目標にFORTRAN、COBOLプログラム例題演習を中心とした電子計算機の本格的な**プログラム講座**を開講する。

プログラミング講座

開講　7月24日　**期間**　夜間6ヶ月間
◆**教室所在地**　国電飯田橋駅5分　家の光会館内
●ハガキ又は、電話で申込しだい**案内書無料送呈**
対象　一般社会人　大学生

日本コンピュータ協会
東京中央郵便局私書箱1830
事務局　☎　0473（22）8082
市川市八幡3の27の9

表6　入出力命令のまとめ

	命　　令	読　み　方	説　　明
入力命令	READ	read	カード1枚分をバッファに読み込む
入力命令	INI　name, *m*, *n*	in　integer	バッファに読み込まれた*m*カラム目から*n*カラム分の整数値を name に入れる。*n*の中にはブランクカラムを含んでもよい。 例： 123456　　INI　A, 1, 6 +123456　　INI　A, 1, 7 □□123456　　INI　A, 1, 8 または INI　A, 3, 6 +00123456　　INI　A, 1, 9 上記4種類のパンチ形式のものを，それぞれ右のような命令で読めばAにはまったく同じ数値がはいる。
出力命令	OUTI　name, *n*	out　integer	name にはいっている整数値を*n*桁プリント。 *n*の指定が実際の数値より小さいと頭のほうの数値が消されてしまう。逆に大きいとその分だけ左側にブランクがとられる。必要なら符号の桁も*n*に含めること。 例：　Aに -0000123456 がはいっている場合 OUTI　A, 8　なら　□-123456 OUTI　A, 6　なら　123456 OUTI　A, 4　なら　3456 とプリントされる。数値が正なら+の代わりに□が一つとられる。
出力命令	OUTH character string	out character	オペランド部の文字列(character string)をそのまま，プリントする。文字の数は72カラム以内なら何文字でもよい。 ここで扱う文字とはMODECで使用できるすべての文字で 0123456789　　ABC……Z +−*/=.,(　)□ の46文字である。 文字と文字の間の□（ブランク）および最初の文字が現われるまでの□は意味をもち，1字として数えられる。
出力命令	PRINT　* PRINT　*n*	print	OUTの四つの命令 OUTI, OUTR, OUTH, OUTBは実際にはプリンタに直接プリントするのではなくて，プリント情報をそれぞれの指定様式に従って出力バッファに移すだけである。1行分がバッファに移されたら，この PRINT 命令で実際にプリントする。 オペランド部の*n*はプリント前の改行数，* はプリントする前にラインプリンタ用紙のページきりかえを指定するものである。

プログラミング　夏期講座　7月1日～8月末
朝・昼・夜

初心者に　ソフトウェア概論
コボル入門
フォートラン入門
アセンブラ入門

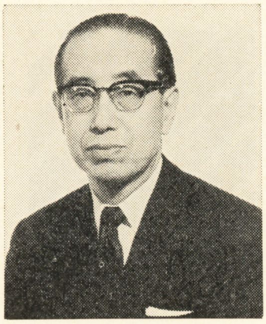
研究者に　数値計算法
解説とプログラミング
技術計算における
プログラミング高度技法

高校生に　コンピュータ基礎知識と
基本プログラミング

講師　大学教官，電子計算機メーカー技術員　多数　　受講生募集中

京都ソフトウェア研究会
〒606 京都市左京区田中飛鳥井町130　　TEL (075) 791−0236

Mr.電算機の低俗さ



飯沢　匡

●ずいひつ

コムピューターと芸術家との関係は今のところ，あまり仲がよいとはいえないだろう。つまり友人としては低俗なやつというのが私のコムピューター君への判定である。

時々，コムピューター氏だか画伯だか知らないが，彼が描いたという大統領の肖像なんてものを見せられても全く下らなくて到底芸術家なんてものの仲間には入れることはできない。

大体芸術なんてものは臍まがり精神の所産と信じている私の目から見るとコンピューター君はこのごろの流行語で申すと「体制べったり」というてあいで，まことに御主人に対して従順で臍まがりどころか忠実そのものである。きっと臍まがり状なんぞ呈したら，忽ち故障ということで，ポンコツにされてしまうからにちがいない。

私の作品は，よく奇想天外という形容詞を頂くのだが，決して奇想は出鱈目でなく，私なりの計算から出ているものなのだ。私は例外が好きだしいわゆる，とりこぼし的現象に関心がある。大勢に逆行するのが私の傾向だがコムピューター君には例外は本質的にないわけだろう。「例」とはプログラムに他ならないのだから。「とりこぼし」はプログラミングがなかったことだし，どういっても私とコムピューター君とは親密になれそうもない。

しかし，テレビ局などはコムピューターが好きで「この時間は，このくらいの予算で，これくらいの出演者で，これくらいの練習時間でできる」と平均値を定めている。何年間のその時間に関するもろもろの数字の集計からそういうものができたのだろう。

こういう考え方は芸術家という臍まがりにとって，ちゃんちゃらおかしい。出演者はたった一人とか平均の何倍という多人数ということも考え出したいのである。装置にしても変ったことをやると，予算は食うこと必定である。試みというのは常に冒険だが，このように枠が定められてしまっては，その枠の中だけの試みで大きな飛躍は望むべくもなくなる。

ことに練習時間なんてものは一人でも下手な役者がいると，いわゆる芝居人がいうところのアンサンブルが生じなくて，非常に手間どることがある。それでもコムピューターに左右されている経営者は冷く「練習は時間切れ」といってくる。つまり平均値の時間から割り出した時間切れである。

もし人間というものが平均値的存在なら，それで結構で，つまり下手な俳優は俳優でない，というのだろうが，世の中には，最初は下手でも稽古に長時間かけると，何ともいえぬ味を出す俳優もいるのだ。コムピューター経営者は，何ともいえぬ味なんてものは認めてくれない。こんなことをしているからテレビドラマなんてものは，どんどんつまらないものになってゆくのである。

このごろテレビ局から身の上調査みたいなものがくる。その紙きれに曰く「汝の得意とするところをできるだけ細く書きこめ」。これはつまり，コムピューターのプログラミングの資料なのだ。

もし，日本で犬が愛されていないと英国人が日本人を攻撃しはじめると，きっと怠者のテレビ局のお係りは，カードを機械にかけて，趣味の項に犬と書き入れをした名士を選び出すという寸法なのだろう。しかし，私は犬は大嫌いで何かというと犬を飼う人に警告を与えている人間だが，テレビ局から回ってきた調書に犬は嫌いなんては書かないだろう。だからあんな調書はたいして役に立たない。もちろん，あなたの嫌いなものという項を設ければいいのかもしれないが，そういう項があったら私は逸早く「コムピューター万能主義」と書くかもしれない。

ずいひつ●

連載　市場調査の話⑤

企業の意思決定と市場調査

門山　允

企業におけるデシジョン・メーキング（意思決定）の必要条件はなにか。電子計算機の導入は，そこでどういう意味をもっているのだろうか…………

1. マーケティング・インフォメーション・システム

市場調査と官庁の調査との大きな相異点の一つは，官庁の調査はどちららかというと事実を正確に把握すること自体が目的であるのに反して，市場調査の場合にはそれが企業の意思決定に直結しているという点である。

企業が意思決定をするためにはいろいろな情報を必要とする。そのうち企業のマーケティング行動のための意思決定に必要な情報の収集・整理・加工・保管・伝達・検索のためのシステムを**マーケティング・インフォメーション・システム**（マーケティング情報システム）という。

マーケティング・インフォメーションというとマーケティングに必要なすべての情報をさすが，マーケティングに必要な情報であっても，それがすべて入手できるとはかぎらず，むしろ必要な情報のうちのごく一部分しか入手できないのがふつうである。

しかもマーケティングの要因の中には不確定的なものがひじょうに多く，たとえ確率的な情報が得られてもそれはあくまで不確定要因であり，確率的な推論を基礎にした予測ははずれる危険性を内包している。それでも確率的な情報が入手できればよいほうで，それすら入手できないため，ラプラス的な無知の原理に頼らざるをえないことが多い。

マーケティング情報の範囲はきわめて広範であり，その内容もひじょうに多岐にわたっている。またそれを収集する方法もさまざまである。しかし，マーケティング情報を収集するための最も有力であり，また最も組織的な方法が**市場調査**であることはまちがいない。市場調査以外にもインフォーマルな情報として人のうわさとか，セールスマンの伝聞だとか，新聞の記事だとか，いろいろのものがあり，フォーマルな情報としても学術雑誌の論文とか，特許公報とか，業界誌の記事などがあげられる。また，よいことではないが産業スパイのもたらす情報などというものが価値のある場合もあろう。

これを概念的に図示すると**図1**のようになろう。この図で各カテゴリーを表わす領域の広さは定量的な意味はないけれども，一応感じのうえでの各カテゴリーのウェイトを表わしており，必要な情報のうちで市場調査という手段により入手できる情報はきわめて限られていることを表わしている。

この概念図からもわかるように，よく「市場調査は役に立たない」とか，逆に，「市場調査をやれば何もかも

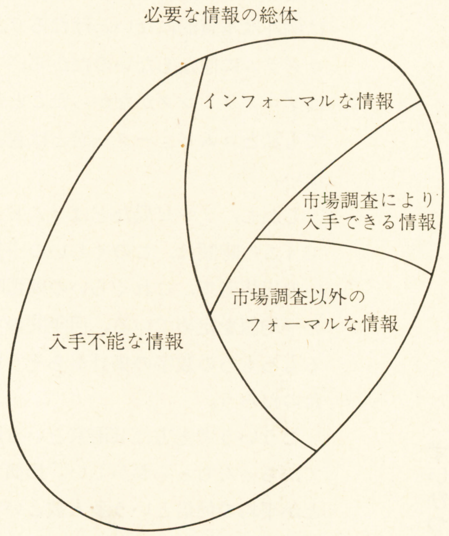


図1　マーケティング情報

Makoto Kadoyama
(株)博報堂

わかる」というような主張はマーケティング情報体系の中における市場調査の位置づけについて無知なためにでてくるといってもよいであろう。

次にこのようなマーケティング情報の情報源をあげると

(1) 官公庁	(9) 問屋
(2) 団体	(10) 小売店
(3) 新聞社・放送局	(11) 顧客
(4) 市場調査機関	(12) セールスマン
(5) 広告代理店	(13) 業務記録
(6) 学会	(14) 社内各部門
(7) 専門家	(15) その他
(8) その他の個人	

があり，それを形式で分類すると

(1) 数字データ（数字）

(2) 文　章
(3) 談　話　}（言　語）

(4) 写　真
(5) 図　面　}（図　形）

というように分けられよう。

この形式は具体的には

(1) 統計	(11) 談話
(2) 報告書	(12) メモ
(3) 公報	(13) 写真
(4) 図書	(14) 図面
(5) 記事(新聞・雑誌)	(15) 録音テープ，レコード
(6) 論文	(16) 映画フィルム
(7) 記録	(17) パンチ・カード
(8) 伝票	(18) 磁気テープ，紙テープ
(9) 帳簿	(19) ディスク
(10) 手紙	(20) メモリー・コア

などの形態をとっている。

これをさらに内容の点からみると

(1) 市場に関する情報
(2) 消費者に関する情報
(3) 販売経路に関する情報
(4) 競争相手に関する情報
(5) 広告に関する情報
(6) 生産技術に関する情報
(7) 営業成績に関する情報
(8) 社会全般の情報
(9) 経済に関する情報
(10) その他の情報

というように分類できよう。

経営管理にEDPの有効的
直結を計る
唯一の総合教育機関

EDP・プログラマー(2級)専修コース

（開　　催）昭和44年10月13日開講～45年３月末
（夜間６ヵ月間）
毎週月～金　夜6.00～9.00
（内容）アセンブラから最新コンパイラに至る教育に十分ウエイトをかけるとともに関連する諸知識体系の有機的結合をはかりながら、プログラミング技術を錬成する、高級プログラマー養成コース。
※　本コースの履習規定に合格した方に、EDPプログラマー（２級）の認定証を授与します。

EDP・プログラマー(3級)速成コース

（開　　催）
（昼間コース　10日）昭和44年７月21日開講
昭和44年９月１日開講
（夜間コース　20日）昭和44年７月21日開講
毎週月～金 9.30～16.30（昼間コース）
18.00～21.00（夜間コース）
（内容）汎用ディジタルコンピュータのプログラミングコーディングの原理及び技術を、本学研究所が開発した教育用アセンブラ言語（SNAP）による演習と機械実習を通して理解していただくコース。
※　本コースの履修規定に合格した方に、EDP プログラマー（３級）の認定証を授与します。

EDP・通信教育コース

☆１年コース　学費　18,000円
３級プログラマー資格認定
☆６カ月コース　学費　8,000円
（基礎コース）
経営管理業務へのEDP適用に重点をおいたコース
※本学および全国主要都市にてスクーリングあり
※入学随時
……………詳細お問合せは下記へ…………………

産業能率短期大学
EDP研究所E係
世田谷区玉川等々力町３－17
TEL（702）4151(代)

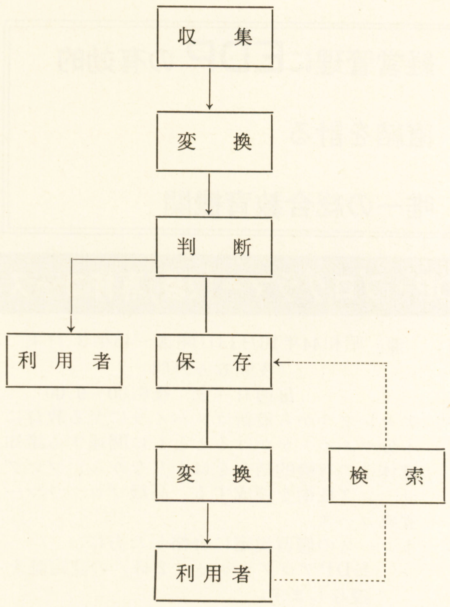


図2　情報のフローチャート

また情報源から情報を入手するチャンネルとしてはまずフォーマルな情報に対しては，どの企業でも

(1) 調査・資料

という機能をもつ部門がある。ただし，これらの機能をもつ部門が名称としてはほかにいろいろのものがついていることがあるが，こういう機能をもつ部門が全然ないという企業は，少なくともある程度の規模の企業にはないといってもよい。

こういう情報収集の専門部門のほかにも

(2) 営業　(3) 製造　(4) 経理

というような現場部門からもいろいろの情報・記録が提供されることはいうまでもない。またインフォーマルな情報の場合には社内のどの部門ということにはかぎらず流入してくるし，またそれは社内に流入してからも不規則な流れをするのが通常である。

流入または収集した情報はそのままの形で伝達されたり，口頭の情報を文書にするというような形態の変換を行なって伝達したり，すぐには伝達しないで一応保存したりする。

情報の利用を容易にするためには単に形態を変換するだけではなく，よりわかりやすい形にしたり，ほかの情報と組み合わせたり，数学的な分析を加えたりするような加工をすることが多い。いうまでもなく，加工は保存のために行なうこともあれば，保存された情報をとり出して利用する段階で加工することもある。一般的にいえば加工は保存部門ではなく利用者が行なうほうが効率的であろう。

情報の保存というのは，古文書の保存のようなものとはちがって，それを必要に応じて短時間で取り出して，利用者に提供できるようにすることであり，そのためには

(1) 形態の変換
(2) 内容の分類
(3) 分類の符号化
(4) 保存の物理的方法
(5) 情報の検索
(6) 取り出した情報の再変換
(7) 利用者への伝達

という各ステップを充分に研究せねばならない。

以上のことを図示すると，**図2**のようになる。

このような情報の保存と検索のためにマイクロ・フィルム，パンチ・カード，磁気テープ，磁気ドラムなどの物理的手段が利用されるようになったことはよく知られた事実であるが，このような意味での**情報検索**（インフォメーション・リトリーバル，略してＩＲ）の問題は現在まだ開発段階であり，インプットされた情報をとり出すだけであればコンピュータがきわめて有力な道具であるが，インフォメーション・システム，とくにマーケティング・インフォメーション・システムにおける最大の問題は情報の収集の段階であり，これはコンピュータを利用しても，労力および時間の節約はたいして期待できない作業である。

2.　マーケティング・システムと市場調査

マーケティング・システムというとマーケティングのための情報システム，社内組織，社外組織，物的流通システム，意思決定システムを総合したものである。言い替えると，マーケティングのための情報，人，物，意思決定のシステムである。しかし，これらのシステムは一つ一つ切り離されたものではなく，実際には一つに重ね合わされているといってよい。

マーケティング要素に直接対応させて社内組織を考えると，次のようになるであろう。

```
                                    ┌調査課
                                    ├企画課
           ┌マーケティング部─┼広告課
社　長─┤                          ├販売促進課
           │                          └営業課
           └他部門
```

この組織はマーケティング上の一つの機能に社内の一つの組織（課）を対応させるという方法で，日本の会社

の場合，基本的にはこの形態の組織が多い。

この形態は，たとえば一つの製品に問題が生じたとき，社内の各組織から横断的に人を集めなければならず，またその製品の売れ行き不振のとき，社内各部門の間で互いに責任のなすり合いになるおそれがある。

これと反対にある一つの製品だけをとり出すと，その製品のためのマーケティング活動は一つのサブシステムになるので，そのサブシステムを単位とした組織が考えられる。このような組織を**プロダクト・マネジャー制**とよんでいる。これは製品ごとに，あるいはいくつかの製品をグループにした製品グループごとに，一人のプロダクト・マネジャー（PM）という職制をおき，そのプロダクト・マネジャーがその製品のマーケティング活動のすべてについて責任をもつ制度である。もちろん一人のプロダクト・マネジャーがマーケティング活動のすべてを実施できるわけはないので，実際にはどうするかというと，調査が必要なときには調査の専門会社，広告については広告代理店，販売促進についても広告代理店，というようにそれぞれ専門会社に仕事をまかせ，セールスマンの活動は社内の担当部門（営業部門）にまかせ，プロダクト・マネジャーはプランニングだけに専念するという方法がとられている。

この制度は前のような組織よりもいろいろの点ですぐれているため，アメリカでは10年ほど前から，日本でもこの２～３年来，一部の会社で採用されている。しかしこの制度の欠点はプロダクト・マネジャーに人材がえられないと効果をあげられない点である。

新製品の開発のときには，このような形の組織ではなく，社内各部門の協力を必要とするので，委員会制をとる会社が多いが，委員会制よりも進んだ形としては**プロジェクト・チーム**という形式がある。これは単なる委員会とちがって，プロジェクト・チームとしての予算をもち，人を採用する権限をもつこともあり，企業の通常のタテ割り組織に対立したヨコ割りの組織であるといえる。

マーケティングのための社内組織には上のようにいろいろの形態があるが，マーケティングそのもののシステムは根本的には同一である。それはマーケティングというものが「消費者の欲する商品を消費者の望むやり方で販売する」ということを原則としているかぎり，マーケティング・システムは（広い意味での）市場調査による消費者の意向をとり入れた，フィード・バックのあるシステムでなければならないからである。

マーケティング・システム全体を解説することは手間がかかるので，そのサブシステムである**新製品開発**のシステムを説明しよう。

新製品開発のためにはまずアイディアの探索というス

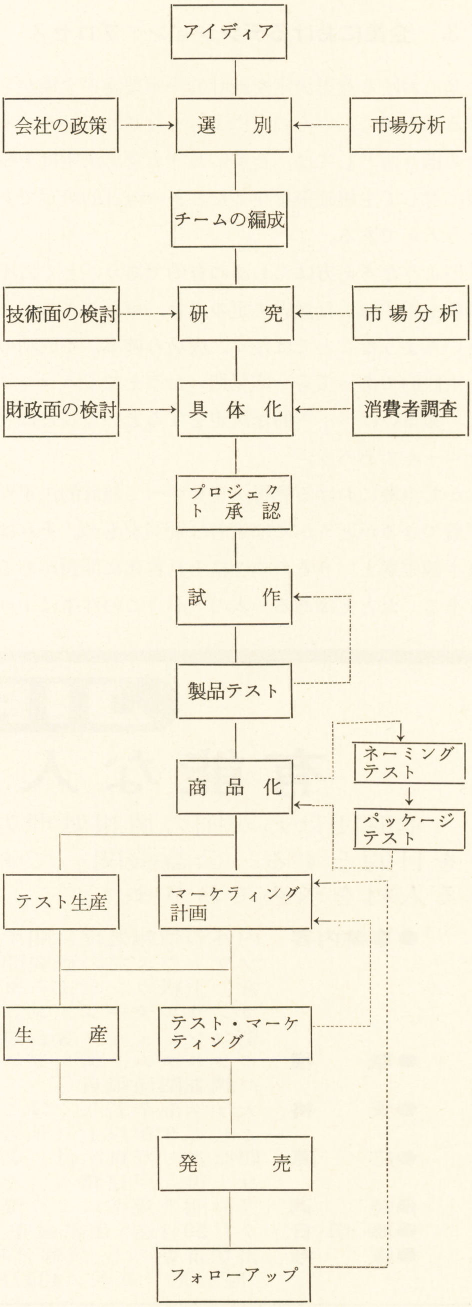


図3　新製品開発の流れ図

テップから始まるが，ここではそれを省略して，アイディアはすでに提供されたものとして説明する。すると，このアイディアをスクリーニングしてから，それが商品化されて市場へ出るまでには図3(前ページ)のような手順で作業が進められる。

3. 企業におけるデシジョン・プロセス

企業における意思決定を統計的決定理論の立場から説明する試みがいろいろの人によってなされている。その場合の拠り所としては，企業における意思決定は不確定事象に対して主観確率を与えたときの統計的決定であるという考えである。

このような考え方はたしかに有効であり，とくに不確定事象を相手にした意思決定の場合，単なるイチかバチかというようなことではなく，成功の確率，不成功の確率を主観的であっても一応見積ったうえでミニマックス的な，あるいはベイス的な決定をするということは合理的なルールであろう。

しかし企業における意思決定をすべて統計的決定理論に帰着できるかどうかには筆者は疑問をもつ。それはやはり主観確率というものの性格それ自体に問題があるからである。たとえばある一人の人が「この仕事は100％成功する」と確信していたとしても，それだけではほかの人に対する説得力はないであろう。妥協的な案として何人かの人にその仕事の成功の確率を予想させ，その平均を採用しようというような提案もされているが，これは主観確率というよりも，心理的な確信の程度が連続分布をしているという仮定のもとに導かれる数量であり，むしろトートロジーであるように思われる。

しかし主観確率とは何かという哲学的な議論はさておいて，ある会社の内部で一定のインフォメーションを与えたうえで何人かの人に，あるプロジェクトの成功の確率を予想させることは，少なくともその企業の内部では整合的な確率系になるとも考えられる。

いずれにせよ，企業で扱う不確定事象の大部分はくり返しのない事象であり，頻度的な考えに基づく確率論は適用できないが，企業にとってはすべての決定が数量的な尺度を基準にして下されることが望ましいので，確率というものが何らかの数値で表わされることが要求される。

たとえば，ある新製品を発売してよいかどうかの決定をするためには，その製品がどれだけ売れるかという見当がつかなければならない。ところが新製品がどれだけ売れるかということは確定的な予想は不可能であり，た

JIPDEC

有能な人材を求める

本財団はわが国の情報処理方式の開発および情報処理産業の振興を目的とする「公益機関」であります。使命感と積極性ある有能なる人材を求めております。

●事業内容　内外の情報処理に関する調査、アプリケーション・システムおよびソフトウエアの研究開発、万国博第二情報システムの開発並びに運営、上級コンピュータスペッシャリストの育成、標準化、共同研究、コンサルテーション、情報処理サービス、普及啓蒙等（最新の超大型コンピュータ３台設置）
●職　種　○システム・エンジニアおよびシニア・プログラマ要員
○調査関係職員
●資　格　大学学部卆若しくはこれと同等以上の学力を有し、実務経験が原則として３年以上（年令35才迄）
●応　募　履歴書（写真貼付）実務経歴書（過去に従事した職務内容明記）身上書、戸籍謄本、その他希望条件等を下記宛提出
●待　遇　当財団の規準により優遇す
●締切日　７月20日迄（秘密厳守）
●選　考　書類審査の上、試験日時、場所をご連絡します（書類は返却しません）
（照会先　東京（434）8760庶務課直通）

財団法人 日本情報処理開発センター 総務部庶務課
東京都港区芝公園２１号地　（〒105）

とえば最初の3カ月で月商1億のレベルには容易に達するであろうが，2億に達するのはむずかしいであろうというような見通しの立てかたしかできない。この場合，容易であるというのは10中8，9であるとか，90%以上の確かさであるとかいう規約を設けておけば，ある人が容易であり，ある人はそうでないというときに，そのレベルを比較できるだけでなく，それをもとにして投資に対する収益の期待値を計算することもできる。

このような確率が主観的なものであるとはいっても，それがある程度整合性をもつことが望ましい。そのためには前提におくいろいろの条件が統一的に与えられ，共通に理解されていなければならない。市場調査というようなものも本質的にはマーケティングにおいて，このような共通の理解をつくり出すための一つの手段であると考えてもよい。

言い替えると，もし市場調査の資料がなければ，ある人はこの製品は確実に月々これだけ売れると主張し，ある人はそんなに売れないと反対したようなとき，どちらも自分のカンだけに基づいてそう主張しているとすれば，その議論は永久に平行線をたどるだけである。これに反して市場調査をもとにすると，必ずしも主張は一致しなくても，その差はより小さくなると考えられる。

4. MISについて

最近日本でやかましくいわれるようになったものに**MIS**(マネジメント・インフォメーション・システム)がある。MISはどういうわけかコンピュータと関連して論じられ，何かコンピュータ以前には経営情報システムというものはなかったかのような議論があるのは珍妙な現象である。企業が活動してゆくためには，意思決定のシステム，情報システムなどが当然存在しているはずであり，それはコンピュータとは直接には無関係である。

企業の意思決定システム，情報システムにはいろいろのレベルがあり，トップ・マネジメントのレベルのシステム，中堅管理層レベルのシステム，現場レベルのシステムなどがある。

現場レベルのシステムは，情報の処理，意思決定などは比較的単純なルールで実行できるが，即時的な実行が要求される。

中間レベルでは，下からの情報を受け取って，すぐにそれを処理（変換・加工・保存）したり，それに対する意思決定をするほか，それを他部門，あるいは上部に伝達する。とくに上部への伝達のときには要約という加工が必要である。中間レベルの場合には逆に上からの情報を受けて，それに対する行動をしたり，その情報を下に伝達したりする役目もある。

トップレベルの場合には，下から情報を受けて，それに対する意思決定をしたり，あるいは行動を指示したりする。

このようなシステムで即時処理が要求されるのは現場レベルであり，中間レベルでは即時処理よりもむしろ情報の加工，それに対する判断が要求され，トップレベルでは情報に対する**意思決定**が主として要求される。

コンピュータを経営情報システムに取り入れた場合，コンピュータの特性として

(1) 大量の情報の処理
(2) 情報の即時的な処理
(3) 高度に複雑な情報の処理

に適していることはよく知られている。

したがって経営情報システムにおけるコンピュータの利用は，現場レベル，中間レベルで有効であり，トップレベルは直接は必要ないともいえる。

たとえば会計情報としてトップレベルに必要なのは毎日の伝票を現場でコンピュータ処理をした損益計算のような要約された情報であり，これは必ずしも毎日必要なものではないし，まして即時的な情報入手の必要はまったくないといってもよい。

ところで，経営情報システムの内容としては

(1) マーケティング情報
(2) 会計情報
(3) 人事情報
(4) オペレーション(生産・販売・サービスなど)情報
(5) その他の情報

というように分類できる。

前に述べたマーケティング情報システムはMISの一つのサブシステムにほかならない。

ところで前にも述べたようにマーケティング情報システムはMISの中でもコンピュータ化がほかのカテゴリーほど効果をあげえない分野であり，MISをコンピュータによる経営情報システムと考えることは大きな誤まりである。

もちろんコンピュータは経営情報システムにおいて大きな役割を演ずることはまちがいないが，トップレベルのための情報システムというものは，トップレベルの個人的な情報受け入れ能力により限定されるので，コンピュータでは問題の解決にならないことは頭に入れておかねばならない。

次回は「市場調査とコンピュータ(完)」の予定

単語帖

ボー

英語で baud と書く。電信線や電話線などの電気的な通信装置によってデータを伝送する場合の信号の伝送速度の単位。

データ通信には，通常は電流の断続，電圧の正負など2準位の電気信号が使われるが，この2準位の信号が毎秒送られる個数をボーとよぶ。ボー数と密接に関連するものとしては，情報伝送速度の単位として使われるビット／秒，すなわち毎秒伝送される2進数字の個数がある。データ伝送では，本来送るべき情報（データ）以外に，通信装置を安定に作動させるために同期信号などの信号も送る必要があり，またデータ通信では誤伝送の確率を減らすために奇偶検査（パリティ・チェック）符号の信号なども伝送する場合が多いが，ボー数では送られるすべての信号を数え，ビット／秒では伝送されたデータの情報だけを数える。したがって，一般にボー数のほうがビット／秒よりも若干多い。

従来わが国の電信では50ボーの信号伝送速度が用いられてきたが，コンピュータ時代のデータ伝送にはこれだけでは速度が不足な場合も多く，またデータ伝送のそれぞれの目的に応じたボー数を選定する必要があるので，データ通信では低速用に50と100ボー，中速用に200と600ボー，高速用に1200と2400ボーが使われる。

バッファ

バッファ記憶装置（buffer memory）ともいう。情報処理システムを構成する動作速度の相異なる部分の間にこのバッファを入れることによって，システム全体の能率を向上させるのがその主目的である。現代の情報処理システムの構成部分中で動作速度が最も速いのはトランジスタまたはIC（集積回路）で作られた中央処理装置であり，主として磁心記憶装置が採用されている主記憶装置の動作速度がこれに次ぐ。

磁気ドラム，磁気テープ，磁気ディスク，ラインプリンタ，カード読取機などは中速度の装置であり，電動タイプライタや電信，電話線によるデータ通信機器は低速度の装置である。

動作速度が著しく異なる装置同志をバッファなしに直結すると，低速の装置の動作が完了するまで高速の装置が待つことになって，システム全体としての無駄時間が増大し，システムの利用効率が悪くなるのである。

バッファには必要に応じて，各種各様の記憶装置が用いられる。たとえば，ラインプリンタを働かせるのに，磁気ディスクバッファ付のシステムでは中央処理装置の計算結果は

磁気コア──→磁気ディスク──→磁気コア──→ラインプリンタと3段階のバッファを通る。

PERT

「パート」と読む。**P**rogram **E**valuation and **R**eview **T**echnique の略で，そもそもは，1958年ころから始まった米国海軍のミサイル開発計画の作業工程管理を合理的に行なうために考案された手法であるが，その後経営計画に広く利用されるようになった。

たとえば，高層のビルを建てるという一つのまとまった仕事をなるべく短い期間に達成するにはどうしたらよいかという種類の問題を計算機の助けをかりて解く手法は PERT/TIME とよばれる。仕事全体を達成するに必要な個々の作業の間には，ある作業が完了しなくては次の作業が始められないという順序関係があるが，何千，何万という作業を含む大きな仕事では最小の所要期間とその期間に最も大きな影響をもつ一連の作業ルート（critical path）を見出すには，コンピュータがきわめて有用である。仕事と作業の時間的解析を行なう RERT/TIME に対して，経費を考慮に入れて，作業工程，人員計画と経費の関係を解析するものは PERT/COST とか，CPM (Critical Path Method) とかよばれる。CPM は，人によっては，上述の PERT/TIME の説明中の critical path に基づく作業工程管理の意味にも使われるので注意を要する。

マクロ命令

電子計算機にハードウェアとして直接組み込まれている命令は，四則演算，条件判定，記憶など比較的簡単で大体1語で1命令程度のものが多い。

これに対して，入出力装置を働かすデータチャンネル（一種の制御装置で中央処理装置と記憶装置を共用する）に入出力装置の動作様式を指示する場合などには何十語分にも相当する情報が必要であり，これをマクロ命令（macro-instruction）という。

また，これ以外にもプログラミングの便宜上，計算機にハードウェアとしては直接組みこまれていない機能をソフトウェア的に計算機に固有の命令と同等に取扱うようにしたものもマクロ命令とよぶ場合もある。

FORTRAN でプログラムを作成する場合，平方根（SQRT），正弦（SIN），指数関数（EXP）などの組込み関数サブルーチンは，使用者から見ると，あたかもこれらの関数を計算するハードウェアがある計算機と同等に見える。これは，ふつうマクロ命令とはよばれないが，その本質は同じであり，マクロ命令とはアセンブラ言語における組込みサブルーチンあるいは組込み関数であるといういい方もできる。

断章

クロード・E・シャノン

喜安善市

情報理論は，送信者より受信者へ伝達される個々の情報の質的内容を問題とせずに，情報を量的に把握し，また統計的に解釈することによって，巨視的な立場からこれを考察しようとするものである。情報理論の創始者クロード・E・シャノンのプロフィールを紹介しよう

1. その業績のあらまし

本誌の読者にとっては，C. E. シャノン（Claude Elwood Shannon，1916～）はあまりにも有名で，改めて業績の紹介をする必要もないと思うが，しかし，現代は変革の時代であり，急速に新人群がつぎからつぎへと現われ，日々世代の交替が進んでいることを思うと，あるいはシャノンはもう伝説上の人物となっているのかもしれない。こんなことを考え，まず，その業績の大要を紹介しておこう。

初期，つまりMIT（マサチューセッツ工科大学）から博士号を得て，ベル電話研究所に入社した1940年代には継電器回路理論（theory of switching circuits）の研究で頭角をあらわした。今次の大戦のためか，その後暗号解読の研究に手をそめ，そこから通信の本質を追求して，情報理論の問題に足を踏み入れた。この業績によって，いまではノーバート・ウィーナー（Nobert Wiener，1894～1964）と並んで，情報理論の創始者としてあおがれている。これはたぶん，戦争末期から戦争直後の仕事であろう。あの有名な論文 "The mathematical theory of communication"（通信の数学的理論）は，1948年の *Bell System Technical Journal* の7月号および10月号に載っているから，その後の仕事は情報理論の完成への努力である。オートマトンをとりあげたのも，情報理論の自律的発展の自然な帰結である。いまから13年ほどまえ，ベル電話研究所から母校の MIT に移ったと聞いている。

Kiyasu-Zen'iti
岩崎通信機（株）

2. 継電器回路理論

現代の青年にとっては，継電器回路理論はブール代数を基礎として理解すべきものであること，したがって論理回路の一章をなすもとして理解することは常識であろう。また，その概念について別に抵抗を感じることもないであろう。しかし，戦前は別であった（ここにも，戦前と戦後の不連続――いまの流行語でいえば断絶――を見ることができる）。

電気通信は電信から始まって電話に中心が移ったが，この電話全盛時代における電気通信工学は，伝送工学と交換工学とに大別されていた。戦前においては，伝送と交換とが対等に扱われるのは概念においてであり，電話企業としてであり，製造業としてで，学問としては別であった。伝送工学は線形数学という数学的基礎があったが，交換工学は数学的基礎が貧弱であった。ただ，トラフィック論に多少の数学的土台があるにすぎなく，継電器回路については，数学的には不毛の荒野であったといってもいいすぎではないであろう。

こんな事情であったから，およそ電気通信工学に志した学徒は継電器回路の基礎づけに情熱をもやしたもので

トラフィック論〔theory of traffic〕
輻湊論（ふくそうろん）ともいう。この問題は，電話交換の問題の通信呼（call）の理論として実際の例と結びついて発達したものである。たとえば，切符売場に切符を買いにきた人が自分より前に行列を作っていて，その人達の用が済んで自分の順番が回ってくるまでに待たなければならないときの待合せ時間はどれくらいであるか，さらに，切符売場のほうからいえば，切符を買いにくる人に不便を感じさせないためには，どれくらいの窓口を用意しておいたらよいか，あるいは行列を見て帰って行く人の割合はどれくらいかというようなことが具体的問題である。

経営の科学

OR

新しい経営技法を紹介するわが国唯一OR誌

オペレーションズ・リサーチ

毎月20日発売　定価250円　年間購読2520円

<特　集>

6月号 未　来　学

日本における未来学の諸様相
朝日新聞論説委員 岸田純之助

未来学におけるORの役割と期待
松下通信 唐 津　一

情　報　革　命　京都大学 坂 井 利 之

〈対談〉未来学における方法論
九州大学 北川敏男　東京大学 近藤次郎

情　報　化　社　会
東京工業大学 林雄二郎　信越化学 山口英治

発展する通信衛星
国際電信電話 松 本　洋

海洋開発の現在と未来
三菱重工業 冨 田 真 己

ビジネスマン・学生の教養書

統計のはなし

●基礎・応用・娯楽— 大村平著 380円〒70円

統計学の初歩を数学の予備知識がなくても理解できるように，マンガや図を豊富に使って，これだけやさしく解説した本は他にありません．楽しみながら，統計の考え方と簡単な手法が覚えられるやさしい統計の入門書です．■絶賛発売中！

ご注文はもよりの書店か，本社直接に願います

日科技連出版社

東京都渋谷区千駄ケ谷1の7の14
電話(402)6511・振替 東京7309 〒151

ある。それらの人々の中に，わが国では中島　章博士と岡田幸雄博士とがおり，アメリカではシャノン博士がいる。継電器の回路に，中島博士とシャノン博士とはブール代数を導入し，岡田博士は位相幾何学を導入した。

中島博士とシャノン博士とは互いに独立にブール代数導入の着想を得たものかもしれないが，完成と発表とはわが国の中島博士のほうがだいぶ早かった。現に，シャノンの論文にも中島博士の論文が引用されていたように思うし，また，ドイツのミュンヘン工科大学のハインリヒ・ピロティ博士から証言を聞いたこともある。さらに，中島博士からもあとで述べるようなエピソードを直接伺ったことがある。

上にも触れたように，1930年代は，伝送工学の発展にエキサイトされて，継電器回路の研究が勃興した時代であった。ドイツでもほぼ同時代にピーチュ女史が中島・シャノンらと同じような研究をしていた。そんな事情で，筆者がミュンヘン工科大学にピロティ教授を訪ねたおり，継電器回路のブール代数的取扱いのプライオリティは明らかに日本の中島にあるにもかかわらず，アングロサクソンは不当にもシャノンしか引用しないことを話し合った。

他方，中島博士(現安藤電気社長，当時日本電気社員)は継電器回路のブール代数論をまとめたころ，たまたま社用でアメリカのベル電話研究所を訪ねられた。その節，研究成果について講演をしたが，後で内容の細部にわたって質問した青年がいた。これがシャノンであった。

このように，シャノンは継電器の回路の研究のプライオリティについては中島に一歩をゆずっているが，この研究で，彼は二進法をブール代数との関連において考察する機会にめぐまれたのである。現代的にいえば，finite mathematics*ないしはディジタル回路的に継電器回路を把握したのであり，これが情報理論研究の方法論となったのである。この意味において，継電器回路のブール代数論は，シャノンにとってはアルファであり，中島にとってはオメガであった。

3.　暗号から情報理論へ

継電器回路理論で確立し，成功を見た研究方法の応用として，また戦時下という社会環境のもとにおいて，シ

* finite mathematics　定訳はないが，しいていえば「有限数学」のこと。つまり，無限や連続の概念を含まない問題に限定して考察する数学で，具体的には，（有限）集合・論理・ベクトル・行列・線形計画法・確率論などをおもな内容としている。

ャノンは第二の研究テーマとして暗号解読を選んだ。このような立場にあれば，シャノンに限らずだれでも研究的野心のある者は暗号の研究に取りついたであろう。当時，暗号は神秘のベールをかぶって，理論的研究のチャレンジを待っていたのであった。

暗号の研究では，通信の本質は何かという反省なしには価値ある仕事をすることはできない。また方法論として，確率・統計論と代数系の知識を必要とする。この意味において，シャノンには情報理論の建設者となる必然性があったといえるかもしれない。

シャノンの暗号に関する研究論文は，情報理論のはじめにある脚註によれば，戦争末期のものであることがわかる。暗号から情報論にアプローチしたところに，シャノンをウィーナーと区別する特徴がある。

世間では，情報理論をシャノン流とウィーナー流とに分けているが，シャノン流とは暗号論的・代数的であり，通信方式的である。これに対して，ウィーナー流は制御論的・解析学的であり，制御系的である。ウィーナーはその著「サイバネチックス」* に述べているように，彼の情報理論は，航空機に対するミッサイルの追尾制御の研究に端を発している。このように，二つの情報理論の流れの相異をその源流において確認することができるのである。

シャノン同とじように情報理論を展開したのは，さきに引用した岡田幸雄博士（当時東北帝大助教授，現電気通信大学教授）である。岡田博士も継電器回路の研究から情報理論にはいった点はあまりにもよく相似している。しかし，二人の理論にははだあいの相異がある。シャノンのには，ストカスチック**な概念が主流であるのに，岡田のにはこれがない。その理由はシャノンが暗号から，つまり電信から出発したのに対して，岡田は電話方式から出発したからではないだろうか。暗号論は電話通信方式論に比べて，はるかにストカスチックな概念を必要とするからである。このように，独立な三つの流れを比較することによってシャノンの立場がよりよく理解されるであろう。

---

*Cybernetics : Or Control and Communication in the Animal and the Machine, MIT Press.

**stochastic 一般には，「確率的」の意。すなわち，偶然に支配される要素を含む体系に関する問題をストカスチックな問題という。

***たとえば，笠原芳郎 著「情報理論と通信方式」 共立出版(1965)，三根久著「情報理論入門」朝倉書店(1964)

****nim たとえば，12個の銅貨が 3，4，5 個ずつ 3 列に並んでいる。2 人のプレイヤーが交互にこの中から何個かずつとっていき，最後に銅貨をとれなくなった者が負けというゲーム。わが国の「三山くずし」に当たる。

オートマトン〔automaton, *pl.* automata〕

オートマンとは元来自動人形とか自動機械を意味することばであった。しかし，今日では次のように考えられている。「情報を受け入れるいくつかの入力端子と，情報を放出するいくつかの出力端子とをそなえた情報処理装置のことをオートマトンという」。

また，オートマトンは次のようにも考えられている。有限個の元から成り，おのおのの元は有限個の状態しか取りえないような機構を考える。時間は不連続的に推移し，各瞬間を整数値 1，2，…，n，… で表わすものとする。元は入力元と内部元とに分けられる。各瞬間における入力元の状態は外部から与えられるが，内部元の状態は，その瞬間以前で，しかもある決まった時間 $n(>0)$ 以内における，それ自身およびほかの元の状態の組合せによって完全に決まるものとする。このような機構を有限オートマトンまたは簡単にオートマトンという。オートマトンの具体例としては，神経網を抽象化したものが考えられている。

シャノンの情報理論の内容そのものについては現在多数のテキストブック*** が出ているから，今さら紹介の必要はなかろう。ただ，シャノンの理論の発展の方向として，符号化の問題とオートマトンとがあることを述べるにとどめよう。この両者ともシャノン以後大発展をとげ，現在なお研究がつづけられている。

4. ゲーム

シャノンは情報理論の自律的発展としてのオートマトンや符号論のほかに，電子計算機との関連において，ゲームの研究をしたことがある。とくに，チェスを電子計算機にささせる研究ではパイオニアのひとりである。このころ，いろいろなゲームを調査したようであるが，1950 年代に室賀三郎博士（現イリノイ大学，当時電気通信研究所員）がシャノンをベル電話研究所にたずねたときにはハノイの塔というゲームに熱中していたという。これはひとりで楽しむゲームであるが，これの数学的モデルは二進法によって最も簡明に記述できる点が，ニム****とならんで興味深い。その点に彼もまた興味をいだいたものと思う。ここでも継電器回路以来，彼は二進法に関心をいだいていたのかもしれない。

■OR入門の決定版！
オペレーションズ
リサーチ入門
近藤次郎著　800円〒70円
オペレーションズ・リサーチ（OR）は　いまや経営者をはじめ管理者　技術・事務・計算機部門そして一般のビジネスマンも常識として知っていなければならない**経営の科学**である．本書はORの起源とその歴史的発展過程にはじまり　代表的な手法の解説とコンピュータおよび他の管理技術との関連にもふれ　OR全般についてその考え方を平易に解説したユニークなORの入門書である．手法の解説には難解な数式は使わず図表を豊富に使って読者の理解を助ける工夫がほどこされている．　■最新刊好評！

統計プログラム入門
あなたもプログラマーになれる
三觜　武著　460円〒70円
管理技術分野に使われている統計手法を取り上げて　電子計算機を使用する場合の計算法　プログラム化への手筋やコツを平易に解説．**はじめて統計計算のプログラミング技法を勉強する人のための独習書　集合教育用テキストに最適**

ORと電子計算機
オペレーションズ・リサーチ誌編集委員会編　1300円〒90円
線型計画法入門
森口繁一著　300円〒70円
計算を中心とした　線型計画法
小野勝章著　1000円〒70円
ALGOL入門
森口繁一編　950円〒70円
システム工学
森口繁一監訳　2000円〒110円

総合図書目録進呈（本誌名明記願います）
日科技連出版社
東京都渋谷区千駄ケ谷1の7の14
〒151 電話（402）6511・振替 東京7309

おわりに

シャノンの情報理論における不滅の業績は他の天才と同様に，彼自身の天賦の才を生かす環境とその中で本質的なテーマを選ぶ本能的嗅覚に負っている。これが筆者のシャノンの業績に対する個人的見解である。

付記　編集部からシャノンの生涯と業績を書くように依頼されたのであるが，多忙のため本文のような著者の個人的感想で責めをふさぐことにした。また，文献など調査不充分である。これらについては読者のご寛恕をお願いする。

編集部注：囲み記事は，矢野健太郎編「数学小辞典」共立出版(1968)による。

ハノイの塔

1板の長方形の板にA，B，C 3本の棒が立っている。厚さが同じで半径が異なる8枚の円板があって，その中央にはいずれも前記の棒の太さよりも大きい穴があけてある。最初にAの棒に8枚の円板が突きさしてある。ただし上のものほど小さい。今この円板を1枚ずつはずして他の棒に突きさす。ただし小さい円板の上に大きい円板を重ねてはいけないという規則がある。こうして円板を1枚ずつあっちへさしたり，こっちへさしたりしてBまたはCの棒に8枚とも突きさし，しかも最初のときのように上ほど小さくなった塔にせよというのである。このとき，必要とする動作の回数は2^8-1である。

一般に，棒Aにn枚の円板が上ほど小さくなった塔のように突きさしてある場合に，上記の要求を達成するために必要な動作の回数をA_nとすれば，$A_n=2^n-1$である。

インドの伝説によれば，ベナレスには世界の中央を指示する大寺院があった。そこのドームの下に3本のダイヤモンドの棒があって，Aの棒には天地創造のときに神が穴のあいた64枚の純金製の円板を突きさした。ただし，上のものほど半径が小さい。そして神は坊さん達に夜となく昼となく，休むひまなく，1枚ずつ円板をA，B，Cの3本のダイヤモンドの棒の間で移しまわることを命じた。円板がAから他の棒に移り終わったときには塔も寺院も坊さん達も皆塵に帰し，すさまじい雷のはためきとともに世界が滅尽してしまうというのである。ところで，$n=64$として

$$A_{64}=2^{64}-1=18446744073709551615$$

仮りに1動作に1秒かかるとし，次々に動作が少しのよどみもなく流れるように行なわれるものとしても，1年を366日として計算すると，583334358156年という年月を要することになる。これなら世界の終わりを心配するにもおよばない。

化学分析の自動化

飯坂譲二

合成繊維，人工タンパク質・新しい合金，われわれをとりまく物質は無数にあり，これまで地上になかった物質さえつくり出されている。現在のラボラトリー・オートメーションの主流でもある化学分析への利用を眺めてみよう

物質の同定，検出やその化学的組成を知るために行なわれる化学分析というと，試薬を加えたり，それを加熱，撹拌したりすることを思い浮べる人が多いかもしれません。試験管やビーカーにかこまれた分析室のイメージとは異なり，近年では，多くの場所に各種の分析機器が設置され急激に普及しています。分析機器は，「物質の構成，種類が異なっていれば，物理的性質が異なる」という，かつては物理の立場からの実験手段であった多くの測定装置を利用したもので，熱，光，電磁気的な性質を解明して化学分析を行なおうというものです（この意味から，機器分析というものを従来の試薬，試験管を用いた化学分析と区別する人もいるようです）。

分析機器のなかには，ガスクロマトグラフのように工業分析として日常的業務に定着しているものもあり，機器分析の省力化としてコンピュータが導入されることもありますが，今回は広く研究のための機器分析の全般についてコンピュータの利用を眺めてみることにします。この世に存在する物質の種類も無数ですし，公害や食品衛生，新しい化学製品の出現など，機器分析は今後ますます盛んになるように見えます。そこで，分析機器の概略を知ったうえで，コンピュータがどこに利用されて，どこにその有利性があるかを考えてみることにします。

1.　分析機器の原理

分析機器には多くの種類がありますが，一般的な物理量を測定の対象とする場合と，結晶構造解析のように，各原子の幾何的配列を問題とする場合とに分けることができると思います（**表1**）。たとえば NMR（核磁気共鳴装置）では，原子核（主として陽子）のスピンという物理量と，その原子核に作用する外場と特定の分子構造によって定まるずれに注目していますし，赤外線吸収スペクトラムは，分子を構成する原子の間の相対的な運動――振動，回転に注目しています。またガスクロマトグラフのように特定の分子内に言及せず，吸着性が物質によって異なっているというマクロな性質を応用したものもあります。

コンピュータの入力という立場からは分析機器の種類はあまり問題となりません。電磁波を利用した分析法では，光源（必ずしも可視ではない）から発せられた電磁波を分析したい試料に当てます。試料に当たった電磁波は試料や波長によって散乱や吸収が起こるので，この変化をうけた電磁波を検出して，スペクトルを測定しています。使用する電磁波が赤外線であれば赤外線分析計ですし，X線を使用すればX線分析計，とくにX線の吸収に注目すればX線吸収分析計となります。また，結晶などの試料によるX線の回折波に注目すれば，X線回折計ということになります。測定されたデータは，赤外線吸収分析計やX線吸収分析計では，波長を変化させた場合の吸収度の変化のスペクトラムとして観測され，X線回折計では，回折角に対する回折波の強度として測られます。すなわち，分析計では，独立に変化させる測定のパラメータ（波長や角度）が独立変数であり，検出した強度のスペクトルが出力となっているわけです。

ガスクロマトグラフでは，吸着剤を装塡した管の一端に分析試料が注入され，キャリアガスによって他端に運ばれる構造になっていますが，到達時間が試料の構成成分によって異なる点に注目して，時間軸に対する検出強度を測定すると，これもスペクトルになります。

質量分析計についても同様です。イオン銃によって加速されたイオンは磁場内を通過する際方向が曲げられ，

Joji　Iisaka
日本アイ・ビー・エム（株）

表1　代表的な分析機器の諸元

	NMR（核磁気共鳴）	X線回折計	質量分析計	紫外線吸収分析計	赤外線吸収分析計	ガスクロマトグラフ
おもな用途	高分解NMR 無機・有機物の同定，構造決定 反応機構の解明 同位元素の定量	結晶構造解析 同定 定性・定量分析	同位元素存在比 反応機構の解明 有機化合物，混合物の組成	有機物の同定 構造決定	有機物の同定 構造決定 高分子化学への利用	無機ガス・有機物の同定，不純物の検出
使用波長域	数十Mc ～100Mc	0.5～2.3Å	（M/e：2～150程度）	0.2～0.4μ	2～40μ	—
分散系	試料自体の共鳴吸収	試料自体の回折	電界または磁界	石英プリズム 回折格子	NaCl, KBrなどのプリズム 回折格子	分離カラム内の吸着剤 不揮発性液体
検出器 出力	短波受信器 0～50mV	シンチレーション・カウンタ G-M管 角度制御4×DO カウンタ 1×DO†	コレクタ （集イオン管） 0～50mVD・C	光電管，光電倍増管 0～50mVD・C	熱電対 ボロメータ 0～50mV	イオン電流検出器 熱伝導度セル
測定所用時間（コンピュータを使用しない場合）	数分	1～2時間	数分～20分	分～時間	10分～20分	数分～30分
変換速度	20点/秒	10秒に1回	20点/秒～45kC	20点/秒	2～20点/秒	10点/秒
前置データ処理	最小自乗法 積算	チェック	ベースラインの補正積算	平滑化	平滑化	最小自乗法 多項式近似

† 4×DO は Digital Output が4個の意味

質量によってその曲率が異なることを利用したものですが，検出器によって測定されるデータは，質量を横軸に検出強度を縦軸にとったスペクトラムとなっています。

このようにみてきますと，分析機器をコンピュータの入力と考えたとき，ほとんどの分析機器からの出力データはその種類によらず，スペクトラムになっているわけです。そして，また分析機器による差は，1)光源の種類・強度，2)散乱・吸収などの強度，3)検出器の特性・性能によって独立に変化させるパラメータの範囲，4)パラメータを一回掃引するための時間，5)スペクトラムを再現させることのできるサンプリング間隔，6)変換精度，7)スペクトラムのピーク形状の特性，などの違いであるということになります。

機器分析の代表的な種類を表1にまとめてみました[1)]。これをみるとわかるように，質量分析計の特殊なタイプを除くと，それほど変換速度の速いものは要求されていません。先回は「ラボラトリー・オートメーション」について述べましたが，そこで，1台のコンピュータが種類の異なった多くの機器を扱うことができるといったのは，この辺の事情を反映させたものにほかなりません。

2.　測定データの読取りとその処理

測定されたスペクトラムは一般にはなめらかな曲線ではなく，種々の原因に起因する雑音を含んだものです。コンピュータと直結していない従来の分析機器では，このような雑音を含んだ信号はフィルタを通し，信号部分をとり出し，アナログ量のままか，あるいはAD変換されたのち記録器へ送っています。しかし，回路的に組めるフィルタの沪波操作には限度があるため，沪波に当たる部分の処理をディジタル計算機によって行なうようになってきました。それが**数値フィルタ**（ディジタルフィルタ）[2)]による平滑化です。

一般に数値フィルタは

$$g_k = a_0 f_k + a_1 f_{k-1} + \cdots\cdots + a_l f_{k-l} \qquad (1)$$

の形で表わされます。ここで f_k は，一定の時間間隔で読み取られたスペクトラムのサンプル記録で，a_{k-l} は重みを表わす係数です。移動平均操作によって SN 比* を改善する方法がありますが，これは(1)式での特殊な場合です。すなわち

$$a_0 = a_1 = \cdots\cdots = a_l = \frac{1}{l} \qquad (2)$$

という特別な場合に当たります（単純移動平均法）。

* 信号（signal）電圧と雑音（noise）電圧の比

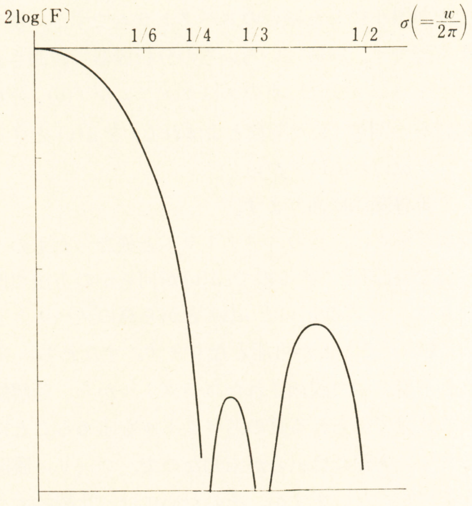


図1　数値フィルターの例

$$g_k=\frac{f_k+2f_{k+1}+3f_{k+2}+3f_{k+3}+2f_{k+4}+f_{k+5}}{12}$$

この移動平均法では，不規則雑音にのみ注目した場合，l を大きくしても $1/f$（f：周波数）程度の雑音に対して効果はうすく，また不規則雑音でない信号部分にも作用するため，系統誤差を招くことになり，この平均するサンプル数の選択が大きな問題となります。あらかじめ信号の周波数特性がわかっている場合，もとのデータに対してフーリエ変換を行ない，既知の周波数領域のみ処理して逆変換を行なうこともあります。前者の計算は比較的小型のコンピュータでも可能ですが，後者はサンプル数にもよりますが大型のコンピュータが必要となります。

1.　**積算平均法**

これは非定常不規則な雑音を除去し *SN* 比の改善を図るために利用される方法です。測定目的とするスペクトラムの掃引を何回もくり返し，同じ独立変数の値のときの測定値を積算することによって雑音を平均化しようというわけです。測定のくり返しは互いに独立な事象ですので，不規則変動の標準偏差が $\sqrt{N}$（Nは積算回数）に反比例している一方，信号分は各回同一と考えられるので，結果的に *SN* 比はくり返しNの平方根に比例して改善されます。このような積算はコンピュータの主記憶装置のコア・メモリを利用します。必要とする容量は分析機器のタイプにもよりますが，1K〜10K 語程度です。

2.　**スペクトラムのデータ処理**[3)]

光学分光計の光源から光をとり出して試料やプリズムに当てる場合，スリットを通して光量や方向が制御されます。解析のために必要なスペクトラムはこのスリット

新刊!!

コンピュータ・プログラミング・シリーズ

プログラマーのための
COBOL入門

D・D・マクラケン　　小岩明，岡祐記 訳

本書は1968年9月報告版の“最新のCOBOL”によって書かれている。COBOL文法の適切な解説と，事例研究による応用の実例が合理的に組み合わされていて，まったくの初心者にもCOBOLによるプログラムのあらゆる問題をマスターできる。また経験者にもCOBOLの一般的な基本概念が理解できるよう配慮されている。演習問題等も豊富に盛りこまれていて，学習テキストとしても完璧である。

B5判・1,500円

好評発売中!

プログラマーのための
フローチャート入門

東和コンピュータ マネジメント

初心者から経験者にまで非常にわかりやすくフローチャートの基本的技法18項目を図解で詳しく解説。

B5判・1,300円

電子計算機　プログラミングとコーディング

P・M・シャーマン　　関根智明 訳

一般的な事項から具体的な方法まで，ごく初歩的な例題から複雑な問題まで広くとりあげた，この分野における最も優れた技術書。　3,700円

竹内書店　東京都港区北青山2－12
振替口座　東京90146

の幅を無限小にした場合のスペクトラムですが，現実には不可能です。一般にいろいろの分析機器で観測されるスペクトラム $g(\nu)$ は，真のスペクトラムを $f(\nu)$，分析機器に特有の関数（装置関数とよばれる）を $a(\nu)$ とすると

$$g(\nu)=a(\nu)*f(\nu)$$

という重畳関係にあります。

観測データは，真のスペクトラムと異なったものですが，系統的な誤差であるので，$a(\nu)$ が何らかの方法で既知となれば，上式の $g(\nu)$ と $a(\nu)$ とから真のスペクトラム $f(\nu)$ を求めることが可能となります。すなわち，限られた分解能の中で観測されたスペクトルから，分解能無限大の装置で得られる真のスペクトルを求めることに対応します。実際には $a(\nu)$ が未知なので，フーリエ展開や逐次近似を行ないますが，そのためには，比較的大きなコンピュータを必要とします。単独の分析機器のみでは望めない分解能が，コンピュータを利用することによって可能になることは，コンピュータ利用のメリットでもあるでしょう。

このようにして得られたスペクトラムも単純な波形ではなく，種々の波形の重ね合せになっています。スペクトル解析で最小限必要なのは，各成分の形，位置，高さ，半値幅です。分析機器からのデータは，通常ガウス形

$$A(X)=A_g\exp\left\{-\frac{(X-X_0)^2}{2B_g}\right\}$$

とローレンツ形

$$A(X)=\frac{A_e}{1+B_e(X-X_0)^2}$$

と考えればよく，また同一の機器の測定結果は同じ型の波形の重なりとみなし，近似波形を

$$G(\nu)=\sum_{i=1}^{n}I_i f(\Delta\nu_i,\ \nu-\nu_i^0)$$

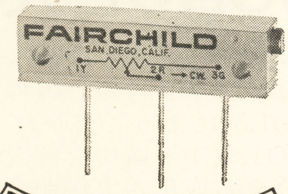
とし　$$Q=\sum_{j=1}^{m}\left\{G(\nu_j)-\sum_{i=1}^{n}I_i f(\Delta\nu_i,\ \nu_j-\nu_i^0)\right\}^2$$

とおいて，Q が極小になるように I_i を決定し，波形の分離を行ないます（むずかしい式が出てきて恐縮ですが，要するに，こういう形の数式を使って処理するのだと思っていただければよいのです）。この種の計算も実のところ比較的大きな計算に属します。

従来の分析機器では，もとのスペクトルの記録上に分離された波形を重ねることはできませんが，図2(b)のようにコンピュータの磁気ディスク装置などのファイルの利用によって，これを重ね合わせてみることが可能となります。

FAIRCON

175kΩ＋175℃
6010p型高温用トリマ



FAIRCON 6010p型トリマポテンショメータは、FAIRCHILD社が新たに開発した高温用、高品質の画期的な、新しい巻線形トリマです。特に密閉性はFAIRCHILD社が絶対の自信をもっております。厳しい環境でトリマをご使用の際はぜひ6010pをご採用下さい。

特長

1. 標準抵抗値で50kΩまで，特殊仕様で 175kΩまで製作可能です。
2. 周囲温度は－55℃～＋175℃です。
3. MIL-R-27208B，グループAの検査を全数行っております。
4. 従来の輸入品と比較して非常に安価です。

仕様

抵抗値範囲(Ω)	10～175K
標準抵抗値(Ω)	100，200，500，1K，2K，5K 10K，20K，50K（100K）
精度	±5％
温度係数	50PPM
定格電力	0.75W　at 85℃
周囲温度	－55℃～＋175℃
環境条件	MIL-R-27208B を凌駕する

（100K）は準標準です。

価格

¥1,500（50KΩ）

株式会社　三和商会
東京都中央区入舟町1-13 ☎(552)6001

SERVO
日本サーボ株式会社

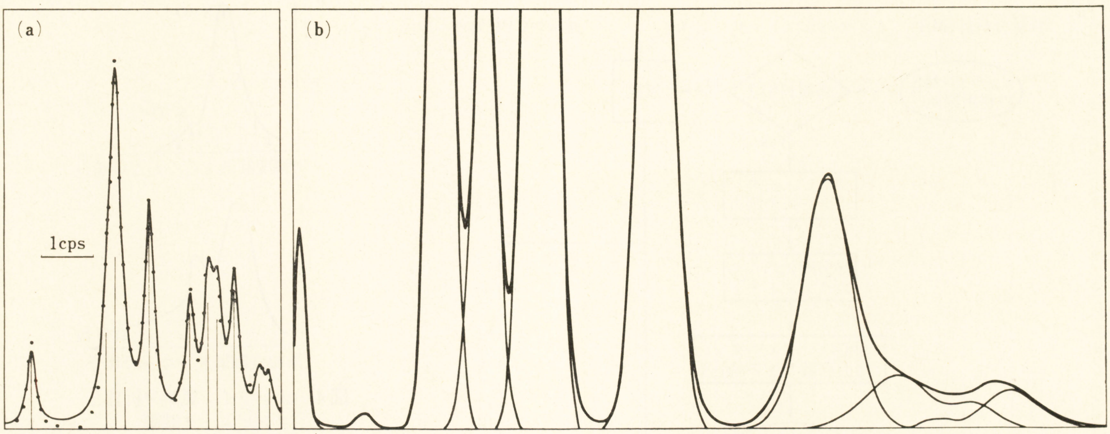
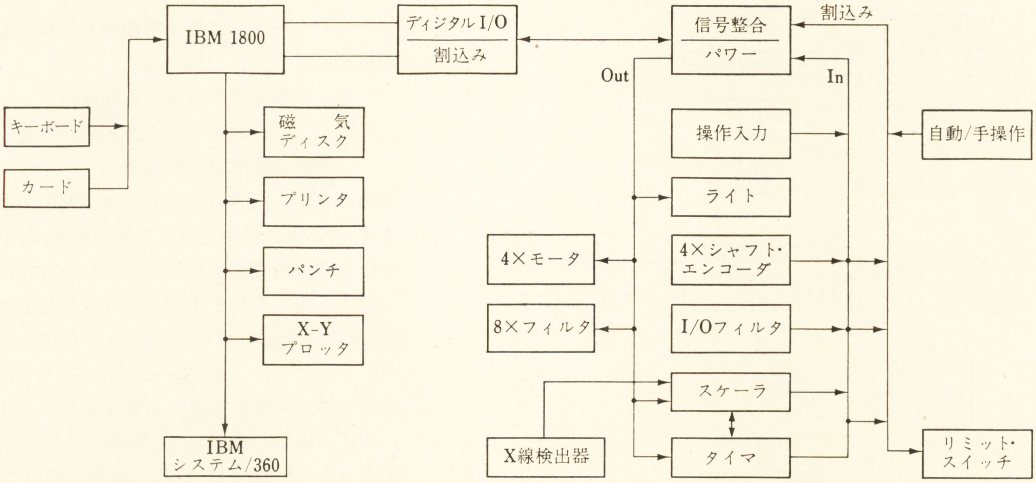


図2　(a)　NMRスペクトルの波形分離（ローレンツ形分布）　点：測定値，たて線：ピークの位置と高さ（波形分離した結果），曲線：計算値
(b)　ガスクロマトグラフの波形分離：太線は分離前の測定曲線



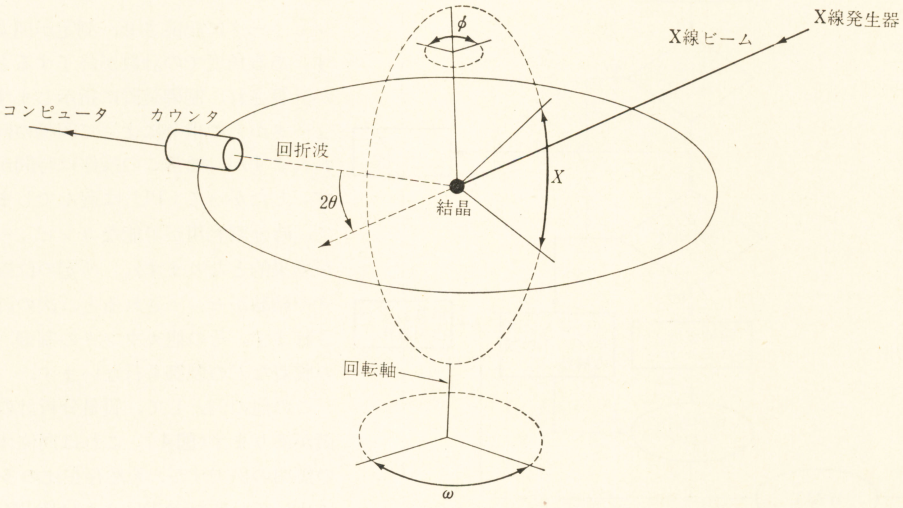
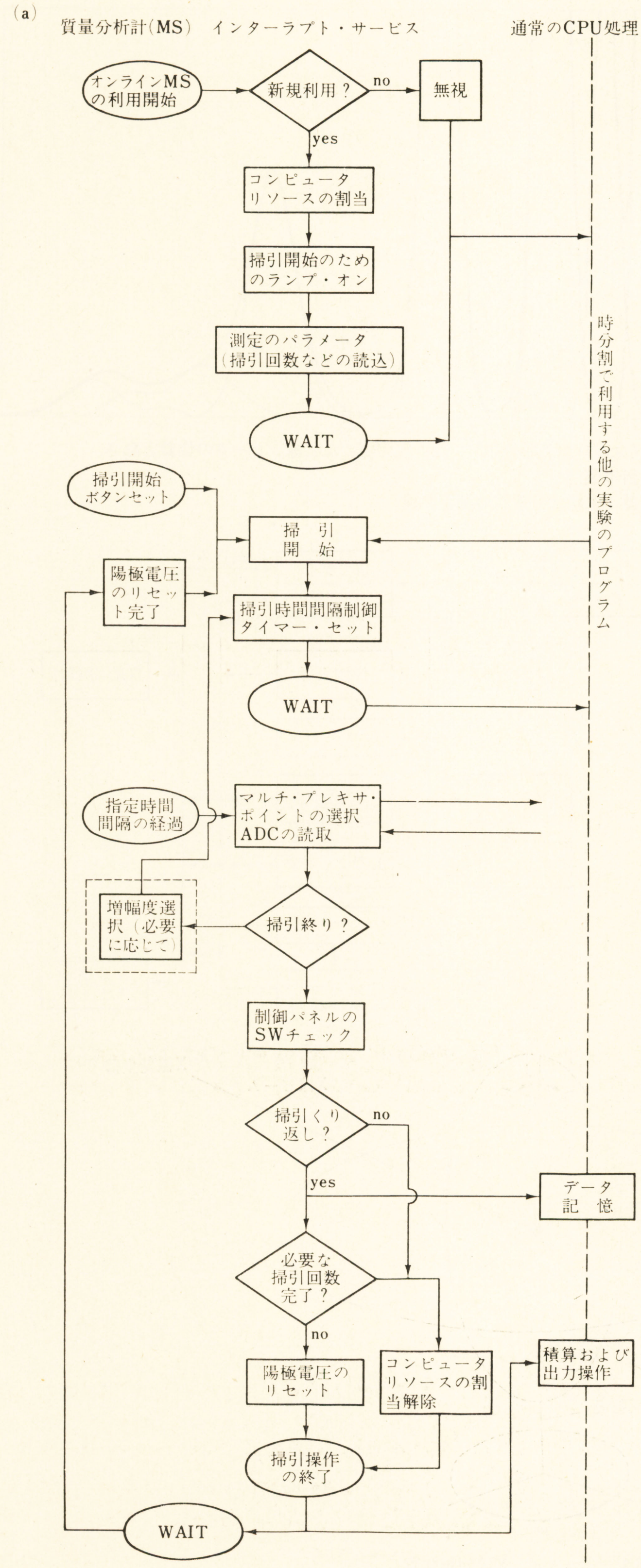


図3　オンラインX線回折装置



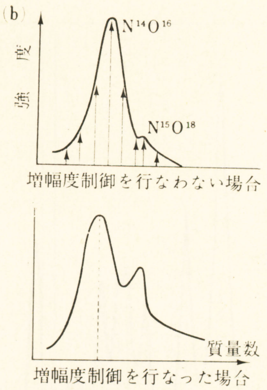


図4 オンライン質量分析計と
コンピュータ制御
(a) 同位体反応の研究における増幅度制御
(点線部分)の例
(b) 質量分析計の増幅度制御の効果(概念図)

3. コンピュータによる分析機器の制御

コンピュータによる分析機器の制御，監視の役割として

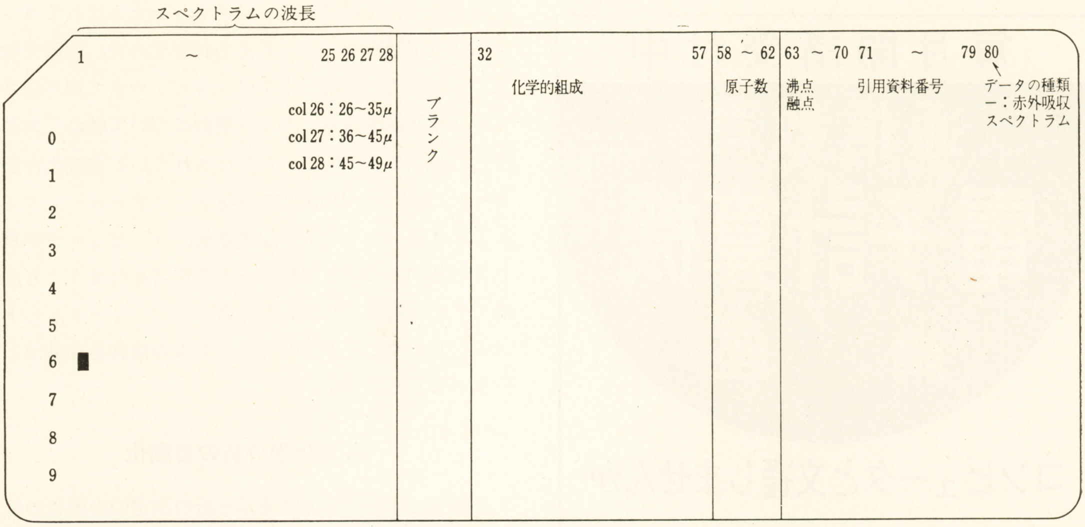
(1) 実験装置の監視，実験保持

(2) プログラムされた設定条件に従って実験手順や設定の変更

があります。前者の例としてはNMRがあります。NMRに使用される磁場の精度は 10^{-8} 程度です。積算法が*SN*比改善のためにNMRではよく用いられますが，繰り返し測定の間NMRの磁場を 10^{-8} に保持しなければなりません。そこで，コンピュータによってNMRの磁場の均一度を監視，調整するわけです。

次に，後者の例としては四軸形X線回折形があります[4](**図3**)。結晶の網面の位置で計数ができるよう，あらかじめプログラムでコンピュータに記憶させ，測定が開始されます。ある角度での計数が終了すると，回転角が計算され，制御装置に指示します(コンピュータからの指令によって四軸の回転が行なわれますが，通常この操作は時間がかかります。したがってCPUは遊んでしまいますので，時分割利用が可能なコンピュータの利用が効果的となります)。所定の位置にカウンタや結晶がセットされると，次の計数が開始されます。その他カウンタの制御，フィルタの脱着などの制御も行ないます。

この他の例として，質量分析計の増幅度制御があります(**図4**)。これは同位体反応機構の解明の例ですが，天然存在比の多い $^{14}N^{16}O$ に比して微量の $^{15}N^{18}O$，または $^{14}N^{18}O$ の部分を検出するため増幅度を変更する流れ図です[5]。



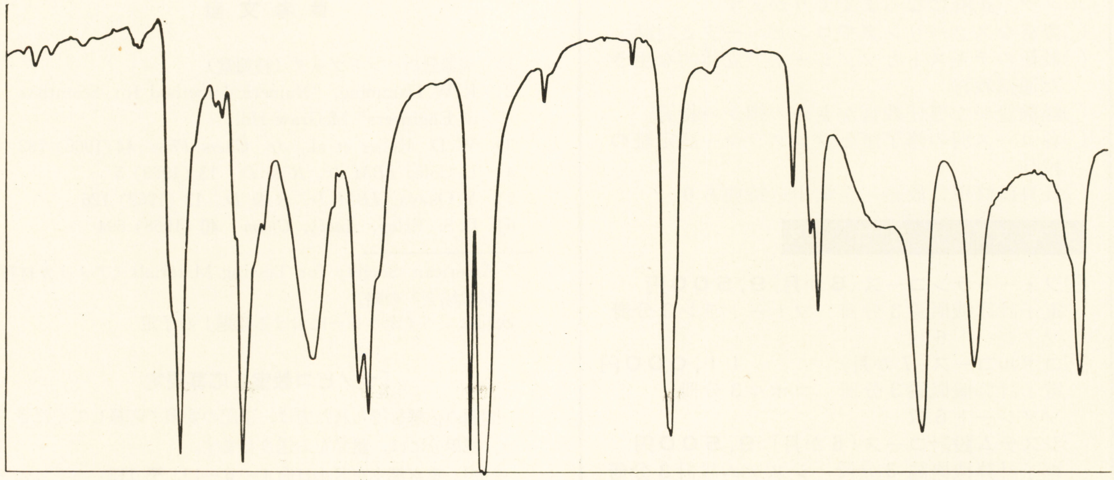
(a) ASTMカード（有機化合物）　　波長域欄の読み方：コラム1の6にパンチがある場合1.6μを表わす

```
INPUT DATA
6.2,6.7,8.7,10.4,11.9,12.9,13.5,14.7,¬(5.5-6.0,7.9-8.4,10.6-11.4,13.2,13.7-14.5,)

NUMBER OF HITS  00004   1-PRINT SERIAL NOS 2-MODIFY DATA 3-PRINT STD DATA 4-EXIT

SERIAL NUMBERS
01314 F   01112 G   00162 H   01535 J
```

(b) スペクトラムの検索入力データ

```
SAMPLE: UNKNOWN CRYSTALS

GROUPS AND BANDS PRESENT  6.2 6.7 8.7 10.4 11.9 12.9 13.5 14.7
GROUPS ABSENT. NO BAND REGIONS  5.5-6.0 7.9-8.4 10.6-11.4 13.2 13.7-14.5
```

(c) 検索結果の出力形式



(d) 赤外線スペクトラム

図5　スペクトラム・データの検索処理（赤外線吸収スペクトルの例）

4. スペクトルの解析

スペクトル測定を行なって，波形の高さ，半値幅面積，位置を知るまでについてこれまで述べてきました。実際に物質の構造を決定し，同定を行なうにはさらに進んだ処理が必要です。結晶の構造解析は，電子分布密度関数を求める必要があり，このためには，フーリエ解析を中心とする比較的大きな計算を必要としますし，NMRのデータの解析もスピン成分がふえると大きな解析計算が必要となります。

毎月開講受付中

通信教育
電子計算機
講　座

コンピュータと文通しませんか

情報革命時代といわれる今日いまやコンピュータの知識なくして将来を語ることはできません。あなたがビジネスマンとして明日を勝ちとるかいなかは、すべてコンピュータを駆使できるかどうかにかかっています。この通信教育は学院内に設置したコンピュータとあなたを結び、親しく文通しながら基礎知識から専門技術まで一貫して学べる画期的なシステムです。

特　色

◯演習問題はコンピュータ（FACOM）がチェック　正解になるまでくりかえす
◯豊富なスクーリングでコンピュータと対話
◯最新のテキストとソノシートで立体的な生きた通信教育
◯経験豊かな専任教官がきめの細かい指導
◯各コース別の修了証とプログラマー認定証の授与
◯社員研修用に最適　企業申込特典あり

テキストと各コース

□フォートランコース（6か月）9,500円
電子計算機概論3分冊　フォートラン3分冊
ソノシート6枚
□コボルコース（7か月）　11,000円
電子計算機概論3分冊　コボル3分冊
ソノシート6枚
□システム設計コース（6か月）9,500円
電子計算機概論 3分冊　システム設計3分冊

◆くわしい受講案内書無料進呈
ハガキでお申し込みください。

日本コンピュータ学院
通信教育部
東京都渋谷区神山町10番地1（郵番150）
電話 東京（03）468-5401（代）

以上二つの問題はシミュレーションとよばれるコンピュータ利用技術の一つと考えられますので，続編で触れることにして，比較的小型のコンピュータで実用化されている赤外線吸収スペクトル解析について触れておきます[6)]。これは ASTM* から発行されている有機化合物のデータ（図5）を IBM2315 磁気ディスク・カートリッジとよばれるファイル上に記憶させ，コンピュータを利用して対話的に必要な標準データの検索を行なうことが可能です。吸収帯の有無を研究者がコンピュータからキー・インしますと，4千のスペクトルの検索を40秒ほどで行なってしまいます。

5.　化学分析の自動化

これまで述べたことは機器分析の測定の自動化の域を出ていないと思います。化学分析の自動化がほんとうの意味で実現されるには，種々のタイプの機器が有機的に働き，試料さえ提供されればコンピュータが分子構造式の決定や定量分析まで自動的に行なう段階にいたって，はじめて自動化の完成といえると思います。それについての試みがあることもきいていますが，これが最終的に実現できれば，研究室ごとの自動化ではなく，研究所あるいはもっと広い範囲を含めたコンピュータ利用の高度化が必要となってくると思います。

参 考 文 献

1)　機器分析ハンドブック（技報堂）
2)　R. W. Hamming, "Numerical method for Scientists & Engineers" McGraw-Hill.
3)　W. D. Kelier et al., *Jr. Chem. Phys.*, **44** (1966) 782
4)　H. Cole, *IBM Jr. R & D.*, **13** (1969) 5
5)　Y. Okaya, *IBM Jr. R & D.*, **13** (1969) 126
6)　D. S. Erley, *Analy. Chem.*, **40** (1968) 894

* American Society for Testing Materials（アメリカ材料試験協会）の略。

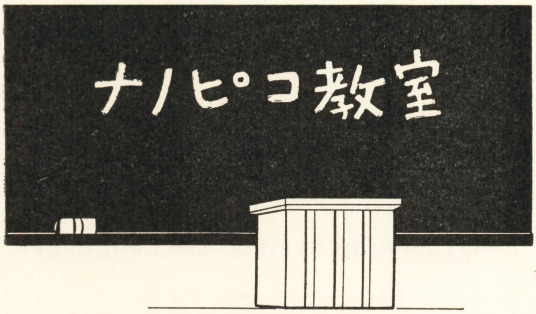
次回は「バイオロジーにおける話題」の予定

「ナノピコ教室」応募規定

問題の正解を得られた方は，下記の要領で応募してください。採用分には，粗品をお送りします。

［送先］東京都文京区小日向4—6—19，〒112
共立出版株式会社，**bit**編集部「ナノピコ教室」係
［締切］1969年7月25日（必着）
［注意］解答用紙は問題ごとに改め，必ず次の事項を付記してください。
1）問題の番号（例：7月号問題1）
2）住所・氏名・年令・職業

なお，各号の正解と解説は3号後の「ナノピコ教室」に掲載します（本7月号の解答は10月号です）。



問題 1 語単位の記憶装置と1個のアキュムレータおよび LAD (load), ADD (add), SUB (subtract), STR (store) などの命令をもつ計算機がある。さて，A番地の内容とB番地の内容を交換したい（データはともに整数形とする）。作業用番地Wを使えば，左のようなプログラムで可能であるが，他の作業用番地を使わずに，この操作を実行するプログラムを作れ。

```
LAD A
STR W
LAD B
STR A
LAD W
STR B
```

注意 プログラマーのかたがたの頭の体操。これができないと，プログラマーのなおれ，というほど深刻に考えなくてもよいでしょう。

問題 2 正の整数の有限個の組 $(a_1, \cdots, a_n)$ がある（n は一定ではない）。$a_1 \geqq a_2 \geqq \cdots \geqq a_n$ の順に並べてあるものとする。このような組に次のように順序をつける。

1. $a_1 + \cdots + a_n$ の小さいほうを若い番号とする。
2. $a_1 + \cdots + a_n$ が同一の二つの組については，そのうちの最大数 a_1 が小さいほうを若い番号とする。
3. a_1 が同じならば，a_2 同志をくらべ，以下順次同じ操作をくりかえして，はじめて違う所が現われたとき，そこの小さいほうを若い番号とする。

このようにすると，$(a_1, \cdots, a_n)$ の組に一連番号をつけることができる。右はそのはじめのほうである。さて $(a_1, \cdots, a_n)$ が与えられたとき，それからその次の番号をもつ $(b_1, \cdots, b_m)$ を作りだす手順を記せ。記述は流れ図でも，操作を示す言葉でもよい。

（番号）	（組）
1	1
2	1 1
3	2
4	1 1 1
5	2 1
6	3
7	1 1 1 1
8	2 1 1

（出題　京都大学数理解析研究所　一松　信）

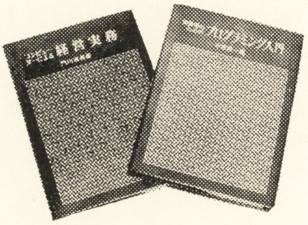
A A+B A+B B B.
B B -A -A A.

「ナノピコ教室」開設のお知らせ　　**編集部**

ナノピコ教室──変な名前ですが，要するに人間のインスピレーションというのは，ナノ秒ないしはピコ秒単位の“刹那”に働くものであるという意味を含めた名前です。一つ，大いにインスピレーションを働かせて，本欄の問題を解いてください（応募規定は42ページ）。

〔EDP入門双書〕

もっとも良心的で親切な使用者の要求にピタリとあった入門双書。著者は機械と現場の両方にくわしいベテラン。



1 文科系学生のための 電子計算機入門　松本隆一著 800円
2 コンピューターによる 経営実務　門川清美著 800円
3 実務家のための プログラミング入門　中原啓一編 700円
4 データ通信とは何か　甘田早苗著 600円
5 中小企業と小型コンピューター　伊藤淳己・西門正己著 700円

※内容見本をお送りいたします。

大阪市北区樋上町45〔電〕363-2531
東京都新宿区神楽坂6の73〔電〕269-1051
創元社

R. L. ワイルダー著・吉田洋一訳

数学基礎論序説

A 5判／448ページ／1, 800円／培風館

1952年に出版された同名の書 “Introduction to the foundations of mathematics” の第2版（1965）の邦訳。

著者ワイルダー教授（R. L. Wilder）の序文にもあるように，本書は彼のミシガン大学での永年にわたる数学基礎論の講義に基づいて書かれたものである。そのような講義を設けた理由は，学部で数学を専攻し，教師その他になって，数学に生涯を託そうとする人たちが，現代数学およびその基礎についての知識を何ももたないままで大学を去ることはよろしくない，と確信したからだという。

本書は2部から成り立っている。

その第1部：数学の基本概念とその方法（Fundamental concepts and methods of mathematics）では，現代数学で広く行なわれている公理的方法や，広く用いられている集合論，それから数の概念や群の概念などがきわめてていねいに説明されている。読者は，これにより，現代数学のおおよその姿を大観することができるであろう。

第2部： 数学基礎論についてのさまざまな立場（Developments of various view points on foundations）では，数学の基礎についてのいわゆる “直観主義（intuitionism）”，“論理主義（logicism）”，“形式主義（formalism）” などの主張が “公平に” 紹介されている。そして著者自身の立場が最終章にまとめられている（この最後の部分は，数学の専門家や，数理哲学者たちの興味をも充分にひきうるものである）。

概評するとすれば，これはけっして “器用な” 本ではない。しかし “使命感と情熱にあふれた” 本である。

訳者もまえがきで触れているように，数学は世人に誤解されすぎている。本書のような書物が普及すれば，このような事態は大幅に改善されることであろう。予備知識としては，せいぜい高等学校程度の数学しか仮定していない。広く読まれることを期待する。訳文はきわめて平明であり，訳書の装訂もなかなかよいと思う。（赤）

Saul Rosen 編

Programming Systems and Languages

McGraw-Hill, 1967, 734 pp.

この本はプログラム言語，コンパイラ，オペレーティングシステムに関する論文選集である。これまでプログラム言語の領域で発表された主要な論文が，この選者 Rosen の奉職していた Purdue 大学の計算機科学科の学生の必読のものと指定され，それらを中心としてこの本がまとめられた。

全篇で35の論文，概説その他よりなり，これが1. 歴史的概説，2. プログラム言語，3. コンパイルと翻訳，4. リストとストリング処理言語，5. オペレーティングシステムの五部に分割されている。最初の歴史的概説は Rosen 自身の筆によるものだが，かなり詳細で，プログラム言語の発展の初期の段階からプログラムの仕事をしてきたものにはたいへんなつかしく，またプログラム言語の講義にはひじょうに参考になる。

そのほか，FORTRAN コンパイラの最初の論文（1957）から IBM のオペレーティングシステムの OS/360, MIT MULTICS にいたるプログラム言語，オペレーティングシステム関係のマイルストーンが網羅されていて，古典的な論文はときどき読みかえしたくなるものだが，その際この本1冊で何度も図書室にいく手間がおおいに省ける。この本を入手したときすでに読んであった論文も少なくなかったがその後この本で読んだ論文もいくつかあり，残念ながら未読のものもわずかに存在する。しかしプログラム言語とか計算機科学とかを専攻する大学院学生諸君はやはり全篇を読んでおくべきであろう。MIT にはこの本を全部読んでないと立入禁止の部屋があるという。1968年アメリカの計算機学会 ACM でまとめたカリキュラム68の報告書でも，いろいろな科目の参考図書としてあげられていた。

この編者 Rosen は Purdue 大学にあったとき PUFFT（Purdue University Fast FORTRAN Translator）をつくって知られている（PUFFT の報告も採録されている）。PUFFT は Edinburgh 大学の Michaelson が「努力しないでソフトウェアに成功する法」のなかで称賛した数少ない成功したソフトウェアの一つである。（和田）

安田寿明著

頭脳会社——システム産業のパイオニア

新書判／297ページ／380円／ダイヤモンド社

情報産業とか，知識産業とかいわれるが，これらの産業は，いってみれば人間産業であり頭脳産業であろう。

科学技術の進歩と，それをバックにした産業・文化の発展は，知識をますます細分化，専門化していく。一方では，情報量の爆発的な増加という現象がある。このような技術と情報のイノベーションの時代の申し子がほかでもない頭脳会社である。頭脳会社は，専門頭脳（知識・技術）の集団であり，もっぱら“考える”ことで商売する。だが，個々の専門頭脳が，いかに深刻に考えても効果はうすい。そこで，頭脳会社は，システムズ・エンジニアリング，すなわち，組織的な方法で，すぐれた頭脳の成果をインテグレートする。

このような頭脳会社が，どのようにして成立したのか，その背景は何か，またその将来にはどのような可能性があるか。筆者は，ランド研究所，スタンフォード研究所，古くはエジソン研究所など，アメリカの著名な頭脳会社を紹介しながら，頭脳会社の現代社会における存在意義について述べている。

ランド研究所が，第二次大戦後のアメリカの戦争技術の温存と平和産業への転用を目的に成立したとはいえ，その後の発展ぶりはめざましいものがある。宇宙航空工学，コンピュータ・サイエンス，経営管理，計量経済，電子工学，宇宙物理，戦略，応用数学，応用物理，社会科学，システム分析の11研究部で，520名の研究者が恵まれた環境のもとでひたすら“頭脳”を駆使しているという。

アメリカの頭脳会社が躍進産業として成長しているのは，一つには，NASAをはじめとする政府機関との間の大規模なコントラクト（契約）に支えられているというが，日本の研究開発機関あるいは頭脳産業志向型企業からみればまことにうらやましいかぎりだろう。

筆者は『知識産業』（共著，ダイヤモンド社）に続いて本書をものにしたわけだが，一貫した問題意識が，新聞記者らしい豊富なデータに支えられて，読む者にせまってくる力作ということができるだろう。　（前田）

★エグゼクティブ・ブックス／新刊

コンピュータの限界

● 道具として使いこなすために

青野忠夫著　新書判／320円・〒70

コンピュータ時代といわれる今日、コンピュータの虚像のみが大きくなりすぎたようである。それは、人間が道具として、奴隷として使いこなすべきもので、振りまわされてはならない。そのために、一般的ビジネスマンが最低限知らなければならないことは何か？——本書は、これまでの入門書にあえて挑戦し、正しい理解のカギを提供する！

情報システムの設計

三重野博司著　A5判／800円・〒90

システム・デザインとはなにか？　それは、いかに実現すべきか？　本書は、経営におけるシステム・デザインの方法を、豊富な実例をまじえ、数式を最小限におさえて、わかりやすく解説した。システム・デザイナーはもちろん、プログラマー、コンピュータ時代に生きぬく経営者・学生諸氏にとって格好の入門書である。

コンピュータ

●機能と応用

J.A.ブラウン著　清水功老／岸田孝一訳

B6判／680円・〒70

本書は、コンピュータについて、その構成、またそれが科学・商業・工業の諸分野で進行しつつあるオートメーションの中で果している役割について、基本的な知識を得たいという読者のために書かれた。図版・写真が豊富であり、最新の情報がわかる。

コンピュータ時代の数学

●技術戦略をリードするもの

渡辺　茂編著　B6判／580円・〒70

これからも著しい進展が予想されるコンピュータ時代に、われわれはどのように数学を使かっていかなくてはならないか？第一線のエンジニアが“役に立つ数学”を平易に説く。

プログラム学習による**COBOL語入門**

J.A.サクソン著　三重野博司訳

B5判／1300円・〒120

事務用計算言語のプログラム学習方式による体系的かつ効果的な入門書である。

ダイヤモンド社　東京千代田区霞が関　振替　東京　25976

Ἐν ἀρχῇ ἦν ὁ λόγος, καὶ ὁ λόγος ἦν πρὸς τὸν θεόν, καὶ θεὸς ἦν ὁ λόγος. οὗτος ἦν ἐν ἀρχῇ πρὸς τὸν θεόν. πάντα δι᾽ αὐτοῦ ἐγένετο, καὶ χωρὶς αὐτοῦ ἐγένετο ⸀οὐδὲ ἕν⸌ : ὃ γέγονεν :2. ἐν αὐτῷ ζωὴ ⸀ἦν, καὶ ἡ ζωὴ ἦν τὸ φῶς τῶν ἀνθρώπων :· καὶ τὸ φῶς ἐν τῇ σκοτίᾳ φαίνει, καὶ ἡ σκοτία αὐτὸ οὐ κατέλαβεν. Ἐγένετο ἄνθρωπος :, ἀπεσταλμένος παρὰ ⸀θεοῦ, ⸆ὄνομα αὐτῷ Ἰωάννης· ¹οὗτος ἦλθεν εἰς μαρτυρίαν, ἵνα μαρτυρήσῃ περὶ τοῦ φωτός, ἵνα πάντες πιστεύσωσιν δι᾽ αὐτοῦ. ¹οὐκ ἦν ἐκεῖνος τὸ φῶς, ἀλλ᾽ ἵνα μαρτυρήσῃ περὶ τοῦ φωτός. Ἦν τὸ φῶς τὸ ἀληθινόν :, ὃ φω-

ヨハネによる福音書
「はじめにロゴスありき」の一節

数理言語学のはなし

水谷静夫

0. 前座的おしゃべり

言語については，だれしもエキスパートである。彼または彼女は，少なくともそのいわゆる母国語を，自在に（と言ってよい流暢さで）あやつる。だからセンスのいい人なら，言語研究者でなくても，その言語に対して鋭い考察を示すこともできる。言語や言語研究にとって，これは幸いであるとともに，ある意味では困ったことでもある。音声学的訓練を受けなくても母国語の発音ができるし，文法論を習わなくても母国語の文をつづることができる。だからこそ言語は重宝なのだ。そうみると，世の言語研究者なんて，ひねくれた表現をいじくり回したり，どうころんでも世道人心には影響のない単語の形を追いかけたり，いずれ閑人の遊びに近いことに没頭していると見えるかもしれない。

そこで，（私を含めて言語研究者から見た）おシロウトさんは，言語研究など無視して勇ましいことをよくおっしゃる。たとえばこうだ。ある数学の先生いわく，単数・複数の区別を形に表わさない日本語は非論理的で，数学には使えたシロモノではない。日本の中学生に4の「平方根」を問うと，2と答えて，－2を忘れる。これが英語なら教師が“the square roots”と言うから，一根だけ答えては不十分なことがわかる，と。これに同意なさる方がおいでかもしれない。しかしそれは違う。文法的数というカテゴリは，あえて論断すれば，論理とは無関係である。

第一に，いまの問が平方根だからよかったが，立方根ならどうだろう。英語だってやはり（平方根を問う場合と同じく）複数形しかないじゃないか。第二に，一般論をする場合に

A dog is faithful.

と

Dogs are faithful.

とどちらでも表わせるというのは，いまの論法では非論理きわまることではないか。もし単複の別が論理の中核にかかわる問題だとすれば，記号論理は当然この区別を記号化するはずである。しかし現にそういう書き分けは行なわれていない。以上のことからして，先の数学の先生の主張は誤りと言わなければならない。さらに，この先生は物を考えるというわざを正当に押えているのか否か，疑わしい。およそ問題を解くということは，解が存在するのかしないのか，存在する場合には，解が幾通りあるかまで含めて，答えることではないのか。うるさく言えば，その前に，問題自体が意味のある問なのかどうかから，見てかからなければならないことだってある。英米の中学生が4の平方根について日本の中学生より正答する率が高いか否かを私は知らないが，たとい高いとしても，複数形でヒントが出されるわけだから，答の半分近くまでは教えられているようなものである。

今は日本語の論理性が直接の話題ではないから，この問題をめぐってさらに立ち入ろうとは思わないが，思い着き程度のことで言語を論じたのでは，的はずれになりやすいということだけは，わかっていただけたろうと思う。一つの言語は一つの構造を担った（と仮定してよい）組織体である。だから，その構造を押えようという態度を抜きにしては，本当の意味での言語研究はできない。

1. 数理言語学とは

構造を問題にするとなると，そういう研究が数学，とくに代数学に結びつきやすいということを，なるほどと思われる読者も多いであろう。あなたがそういう方なら話がスラスラ運ぶ。しかし，読者が皆さんそうとは限る

MIZUTANI Sizuo
東京女子大学文理学部

まいと思うので，少々のまわり道も必要であろう。

以前の言語研究で数学を使うと言えば，ほとんどすべて統計の利用であった。言語現象には統計的に決まる面があるので，統計的研究が言語学の中心に触れないとは言い切れない。この点，文法論の構築において統計的見地を否定したチョムスキの見解は，純粋性を追って狭きに失したと評せられるのであろう。しかし，彼の主張にも一理あって，統計的見地だけでは言語の構造は明らかにならない。ここ十年ほどの間に言語理論は（チョムスキ流のものに限らず）著しい代数化を見せてきたが，その動き自体がこういう事情を物語っている。「数理言語学」の名には後者の研究のほうが似つかわしいかとも思える。

わが国には，会員数は少ないが計量国語学会というのがあって，その創設は1956年末であり，この種の学会としては世界でも古い(！)ものである。創立準備に当って学会の名を何とするかには，だいぶ議論があった。「数理国語学会」も候補名の一つだったが，当時としては木に竹をつぐ感じを与えるに違いないということで，まだしも統計と連想がしやすい「計量国語学会」を選んだが，会名の英訳のほうでは，将来の領域拡大を見込んで“mathematical linguistics”を使った。数年ならず，主として機械翻訳からの刺激により，われわれが予想していたような学会が世界的な規模で結成される気運を生じた。そして今日では「計算言語学(computational linguistics)」の名がかなり一般的になっている。この名で人がどんなものを描くかは，なおまちまちのようである。ある人はコンピュータと結びつけて言語データ処理の研究だと割り切るだろうし，他の人は「計算」ということばの日常的意味からして統計的研究を思い浮かべるだろう。この夏ストックホルムで開かれる計算言語学国際会議の組織委員の一人が配った招待状には，この学問の概念規定への言及がある有様である。しかし，いずれアルゴリズムを中心問題とする言語研究へとまとまってゆくような気がする。本誌編集部の依頼した「数理言語学」は，ほぼこの計算言語学に当ると考えてよかろうか。

しかし，こんな調子で外面的な紹介ばかりしていても始まらない。二つほど話題を取り上げて，内容の一端をお伝えすることにしよう。

2. ルーンの自動抄録法

ドクメンテーションや情報検索（information retrieval, IR）に関心のある方なら，一度はどこかで故ルーンの自動抄録法に触れておいでであろう。コンピュータによる論文抄録作成技法の開発が必要なことは，いまさら改めて述べるまでもあるまい。その「古典的」手法がルーンの方法である。自動抄録も機械翻訳と同様，出発時の花花しさにそぐわず，このところ足踏み状態にあるが，それはともかくとして，ルーンのアイディアを大づかみに述べれば次のようになる。

原報を走査して各語の使用度数の統計をとる。どんな文章を書くにもいる基本語（たとえば，the, is, for など日本語でなら格助詞や，スル，コト，ナル，イルなど）を除いた範囲で，その論文で使用度数の大きい語を選び出す。そういう，いわばキーワードを多く含む文を見つける。そうした文を適宜の数だけ，原報に現われる順に並べたものをもって，その論文の抄録とする。

こうして作った抄録の文章は，人間抄録に比べればぎくしゃくした行文になり勝ちではあるが，その道の人が読めば優に人間抄録の代用になる場合も多いし，いろいろな点で長所があるのも事実である。もちろん，欠陥も多い。そのことにはあとで触れるとして，このルーンの方法は全く統計的見地から考え出された技法である。だからと言って言語の性質に無関係なわけではない。まずこの点を見ておこう。

筆者は以前，国立国語研究所で雑誌の語彙（ごい）調査に従事していた。用語というのは思わぬところで社会を反映するもので，数字のような一見ニュートラルなものを取っても，雑誌のジャンルの推定がつくことがある。昭和20年代の婦人雑誌には，0.3だとか，1.5だとか，30だとか，とにかく3で割り切れる数値が多い。これは和裁記事で一分・五分・一尺をメートル法に換算したからである。最近の婦人雑誌の用語統計ではどうなっているか知らないが，おそらく，和裁記事が洋裁記事に圧倒されたことからして，こういう特徴は消えてしまったのではあるまいか。

また，甲・乙二つのジャンルの雑誌に関する用語統計表があるとする。甲では数値語全体の使用率があまり大きくなく，個々の数値語は億の台のように大きい値を表わすものが多く，逆に乙では数値語使用率が割合に大きく，個々の数値語は細かい値を表わすものが多いとする。このことだけからでも，甲は総合雑誌，乙は婦人雑誌と推定してほぼ間違いがない。総合雑誌は案外に数値に立つ記事が少なく，数値が引かれるときには国家予算などを話題にすることが多いというような実情による。婦人雑誌のほうにコメントをつける必要はあるまい。こうした事情があるから，コンテント・アナリシスという技法も役に立つのだと思われる。

さて，こういうわけで，ある程度長い論文の用語統計

表を作れば，次の傾向がはっきり認められる。

使用度数の大きい語は，(1)どんな内容の文章を書くにせよ文の骨組を形作るのに使わざるを得ない共通基本語か，(2)話題の分野での一般的な語（たとえば，ソフトウェア関係の論文中の「計算機」「OS」など）か，(3)その論文の主題に関係の深い概念を表わす語か，である。

実際上は(1)と(2)，また(2)と(3)との境目がはっきり引けるとは限らず，また順序関係を論じた数学論文での「前」のように(1)と(3)とが紛れる場合もある。しかし

その論文の主題と密接に関係する語は，その
論文において使用率が大きい

という命題は，それが真である確率の大きい命題である。それにしても，たとい共通基本語を除いて考えてもこの逆命題が真である確率は原命題に対する場合ほど大きくはないようである。

ルーンの方法は，彼がこうした言語学的知見に支えられていなかったとしても，まさに上記の用語法則を利用したものだといえる。ただし，その用語法則は十全のものではないから，ルーンの方法では役に立つ抄録が得られないことも起る。そこで，ルーンの方法の改良が二つの方向で図られる。一つは，抄録に採用する文を選ぶ規準値に関する，いわば物さしの改良である。これは当然キーワードの定め方から始まって，現象の数量化の問題，決定関数の作り方の問題となるが，直接には言語の問題ではないから，これ以上立ち入らない。もう一つは，言語学的知見をもっとふんだんにとり入れる方向での改良である。ルーンの方法では，文法も意味も，文章全体の結構も利用していない。それゆえ，極端な場合には，原報のある部分が，「AはBだ。一般にはそうだが，Aに伴ってしばしばCが起る。そのときにはAはBでない」という形であるとして，そこから始めと終りの文とが採用されて，抄録文の中に「AはBだ。そのときにはAはBでない」のような部分が作られてしまう可能性もある。これを統計的に防ぐ手立てはあるまい。意味を利用しないまでも，何らかの連糸操作（string manipulation）を施すような方法によらざるを得ない。

ついでながらルーンの方法の改良策として，すぐれたものは，まだ出ていないと思う。

ともあれ，筆者は統計的研究が言語の本質を全く解明できないとは考えないが，それにしてもそういう迫り方だけで（あるいはまた，あとで述べる代数的な迫り方だけで）言語の問題が扱い切れるものではない。そこで，定量的な統計的研究にあわせて，いわば定性的な研究が必要になる。

3. 文法論の代数化

言語研究と聞けば「ああ文法か」と考える人が多いし，そうでなければ，とくに国語学の場合，国語教育の研究と混同する人もいる。これは言語研究者にとっては迷惑なことであった。だが，最近のように構文論花盛りの気運では，言語学とは構文研究のことだと思われても多分それは的はずれな見方であろうが，仕方がないとも言える。また国語教育で漢字の問題の重みがかなり大きく，一方ティーチング・マシンのこと，映像表示装置の発達などを考えれば，純粋に国語学的な立場からの字体研究も，こうした問題と結びついてゆかざるを得ないかもしれない。文字論とて数理言語学の一領域になり得る（余談だが，私はリスト構造を利用した漢字処理に関心をもっている）。こうして言語のいろいろな問題が形式理論として再編を受ける方向にあると言える。意味論でさえ形式化の試みが，日本でも外国でも始まった。しかし中でも最もその動きが際立っているのは，文法論であろう。

1956年，ソ連の女流数学者クラギナが集合論を利用した文法論を発表した。翌57年には米国の言語学者チョムスキが *Syntactic Structures* を刊行して変換文法を提唱した。この時期から，数学的または数学風な装いを施した文法論が世界の各地で続々と発表された（日本でのこの種の研究のスタートも同時期である）。これらの中のいくつかは独立に企てられたものであり，そういう動向のおもな原動力が機械翻訳での必要にあったことは否めない。今日，フレーズ構造文法を基底として変換を使うチョムスキ流の生成文法が優位にあるが，チョムスキ自身は（おそらく）機械翻訳にはあまり興味がないように，こうした動向を何でも彼でも機械翻訳に結びつけてとらえるのは，偏狭に過ぎよう。

日本では指摘する人がほとんどないと言っていい有様なので（もっともアメリカでも言語学者はそれをさほど強く意識していないようだが），以下では，フレーズ構造文法を中心に，それを記号論理やオートマトンの研究の発達の流れの中に置いて，ながめてみよう。フレーズ構造文法はプログラミング言語，その他の形式言語とも縁が深い。なお，紙幅が乏しいので，かけ足で進むより仕方がないが，もしこうした話題に興味を覚え，少し詳しく知りたいとお感じの読者がおいでなら，『国文学解釈と鑑賞』（至文堂）の本年五月号から七月号まで連載する拙稿「思考と言語」を見ていただきたいと思う（そこには参考文献もあげてある）。

4. フレーズ構造言語の定義

初めに，トリヴィアルながら問題の核心に触れる例を出そう。「自然数」には0を含めない場合（つまり正整数）と含める場合とがある。この二通りの自然数の十進表示の形を作る文法を考える。読者にはバッカス標準式*を御存じの方も多かろうから，それで書けば，初めの意味での自然数，すなわち正整数の表示に対して

〈正整数〉::＝〈有効数字〉|〈正整数〉〈数字〉

〈数字〉::＝0|〈有効数字〉

〈有効数字〉::＝1|2|3|4|5|6|7|8|9

はそれを生成する文法（の一つ）である。たとえば“204”は，まず“2”が第三式の右辺二番目の項からして有効数字，したがって第一式右辺第一項からして正整数であることがわかる。また，第一式の右辺第二項により正整数の直後に一つの数字を添えたものも正整数であるが，事実“0”が第二式右辺第一項により数字であるから，“20”もまた正数数である。同じ考え方で，この“20”の直後に数字一つ添えたものも正整数であるが，“4”が第三式・第二式によって数字であることが確かめられるから，“204”は正整数である。こういう調子で，正整数の意の自然数（の十進表示形）はみな，前記の規則の範囲で作り出せる。自然数は無限に多くあるから，それを表わす形も無数にある。しかし前記のような有限個数の規則だけで規定されてしまった。次に，0を含めた自然数に対しては，

〈自然数〉::＝0|〈正整数〉

という規則を置き，あと前記の三つの式を追加して規定するのが，手っ取り速い。ここで注意すべきは，“204”が二百四を表わすか，二十四を表わすか，それとも四百二をかというようなことには無関係に，こうした文法規則が定められているということである。つまり，記号列である連糸の形にだけ着目して理論が構成されている。

そういう立場では，整数

……3, 2, 1, 0, −1, −2, −3, ……

の全体を，次の規則の集合で作り出せる連糸がさすものの集合というようにも定義できる（実はこれだけでは，数学上の対象たる整数の集合を完全に定義したとはいい切れないが）。すなわち

〈整数〉::＝〈正整数〉|0|−〈正整数〉

〈正整数〉::＝〈有効数字〉|〈正整数〉〈数字〉

〈数字〉::＝0|〈有効数字〉

〈有効数字〉::＝1|2|3|4|5|6|7|8|9

このような規則の組を次の形式で表わすこともある。

整数	→	正整数
整数	→	0
整数	→	− 正整数
正整数	→	有効数字
正整数	→	正整数 数字
数字	→	0
数字	→	有効数字
有効数字	→	1
有効数字	→	2
有効数字	→	3
有効数字	→	4
有効数字	→	5
有効数字	→	6
有効数字	→	7
有効数字	→	8
有効数字	→	9

どちらの表わし方をするにせよ，0から9までの各数字とマイナス符号とを単語と見なし，「整数」を「文」と見なし，「正整数」，「数字」，「有効数字」を構文要素の名と見なせば，前掲の規則は文法法則を記述したものだと言える。そして{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, −}は，整数連糸つまり文を形作る（末端）語彙，また{整数，正整数，数字，有効数字}はいわば文法術語の集合，すなわち非末端語彙である。正式には，末端語彙と非末端語彙とをあわせた集合が「語彙」と呼ばれる。文法法則に当る$\alpha \to \beta$の形のもののαとβとは，この語彙に属する記号から成る有限連糸であって（ただし，一般論としては矢印の右辺は空連糸でもよい）また純粋に形式的には矢印を消してしまい$\alpha \to \beta$を順序対（つい）(α, β)と考えてもよい。

以上を準備として，自然言語・形式言語のどちらに関しても共通に，フレーズ構造文法を形式的に定義することができる。すなわち

書き替え系，またの名**フレーズ構造文法**の同義語としての一つの文法Gとは，次のものの四つ組(V, T, P, σ)のことである；ここに

Vは，**語彙**と呼ばれ，原始要素たる**語**を元とする空でない有限集合；

Tは，**末端語彙**と呼ばれ，Vの真部分集合（Tの元である語を「末端記号」という）；

Pは，**作り出し**とか**書き替え規則**とか呼ばれる順序対(α, β)を元とする有限集合であって，αは「非末端記号」つまり非末端語彙V−Tに属する語を少

* 「::＝」は，その左辺のものが右辺のように定義できる意。右辺の「|」は「または」の意。〈　〉はそれに包まれた名で呼ばれる一記号をさす。こうした約束で書いた式がバッカス標準式である。

なくとも一つは含み，かつVに属する語から成る有限連糸，βは空連糸をも許すとしてVに属する語から成る有限連糸；

σは，**当初**(初期)**記号**と呼ばれ，V−Tに属する特別に指定されたただ一個の語

である。

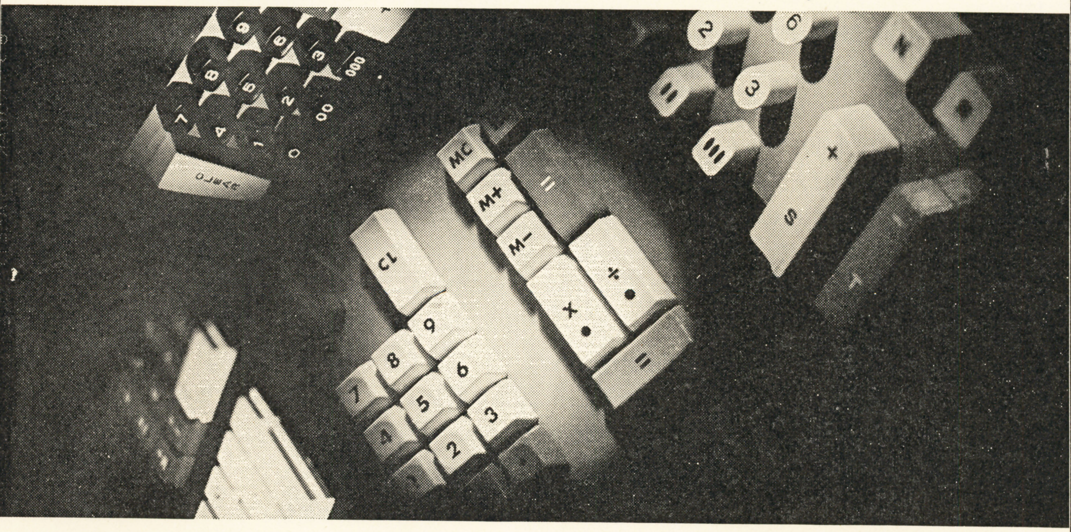
先ほどの例では「整数」がσとして指定されている。なお書き替え規則には，ふつう，さらに細かい制限を設けるが，今はそれを述べることを省略する。この定義の見掛けはおっかないかもしれないが，先ほどの例と見比べて落ち着いて読めば，それほどむずかしいことを言ってはいないことがわかるであろう。

これで一往，「文法」の定義ができた。では「言語」はどう定義されるか。略式ではあるが次のようになる。まず，この形式体系Gで，当初記号σは唯一の公理である。公理だから，議論をするときにはσを真だとして前提にすえてよい。次にPの中で矢印の左辺がσである規則，つまり $\sigma \longrightarrow \beta$ の形のもの（の任意の一つ）を取り，σをβに書き替える：このことを $\sigma \Longrightarrow \beta$ と書き表わす。βは一記号かもしれないし，二記号以上から成るかもしれない。また一記号だとしても，それは末端記号かもしれないし，非末端記号かもしれない。もし，βに非末端記号が現われていれば，その任意の一つを含む適当な部分連糸（むろんその記号だけでもよい）を取ってそれと同じ形の連糸を矢印の左辺にもつ規則がPの中にあるか否かを調べる。もしあれば，そういう規則（の任意の一つ）によって，左辺の形を右辺の形に書き替える。こういう手続きを繰り返して $\sigma \Longrightarrow \beta \Longrightarrow \cdots \Longrightarrow$〈末端記号だけから成る連糸〉となれば，最後に得られたその「末端連糸」を文法Gによって**生成**された**文**と言う。こうした文の集合 L(G) が文法Gによって生成される**フレーズ構造言語**と定義される。あるいは略して，単に「言語」とも言う。

こうした定義を聞くと，話が逆ではないかと言う人が出てくるかもしれない。文法が定まってあとから言語が決まるというのは，ALGOL のような人工言語ならそれでもいいだろうが，日本語や英語のような自然言語はその自然言語が先にあって，それについて国文法なり英文法なりが発見・整理されるのではないか，と。認識の順序としてはそれに違いない。しかし，そのことと理論の整備とは異なる。自然数に関する公理論ができはじめて自然数が発見されたのではないのと，事情は同じである。

またもし，それはわかったが言語の定義はそういうも

二色成形品キートップ



◈営業品目…圧縮・射出・二色成形品/プリント配線板/コネクター/電子計算機部品組立

山下電気株式会社

本社 東京都品川区南品川 3—6—33 (474)—2711

のしかないのかと問われたら，今述べたのは一つの定義に過ぎないとお答えしよう。ある種の問題ではそう考えるのが便利だからそう定義するまでのことで，私自身，自然言語に対する別様の問題では別様の定義を採用する。

5. 形式体系としての言語

さて，前の節で述べた書き替え系は，チョムスキが変換文法の基底として提案したフレーズ構造文法を形式的に整えたものである。ところで，こうした考え方は，何もチョムスキによって突然唱え出されたわけではない。たとえば，彼の *Syntactic Structures* よりは七年早く刊行されたローゼンブルームの *The Elements of Mathematical Logic* を開いてみる。その第四章には，「言語の一般構文論」という題がついているが，初めの部分に次の文言が見える。

> 言語は，ある種の記号およびこれらの記号の連糸から成る。その構文論はこれらの連糸の分類と変形とのための諸規則とから成る。言語の字彙は，ふつうには有限種類の，ある基礎的記号から成る。連糸という語で有限の長さの記号列を意味する。…われわれは英語を次のように定義していいかもしれない。その字彙は“*a*”,“*b*”,…,“*z*”の文字，通例の句切り符号，相続く語の間の空白のための記号から成る。語はオクスフォード英語辞典に収めてある任意の連糸である。文はある標準英文典の規則に従って形作られた任意の連糸である。……

そして，自然言語は構文法が複雑でしかも例外も多いから，数学者や論理学者は，もっと単純でしかも規則正しい構造をもった一種の人工言語で仕事をするのだと言っている。また付録には，論理体系三つに対し，フレーズ構造文法と見紛うばかりの形の規則などを，字彙・公理・作り出し・解釈の順に並べて示してもいる。

記号論理のこの種の扱い方は「組み合せ系」の名で知られるものであるが，公理（一般には，無論，一個とは限らない）から出発し，別に指定された推論の規則にのっとって，作り出しを使って，公理である連糸の書き替えを行なう。そうして得られるものが「証明可能な連糸」つまり「定理」である。こう見てくると，チョムスキの文法理論はまさに，変換文法を含めて，記号論理が担う形式体系での取扱いを模写したとさえ見えてくる。彼自身，その「生成」の概念を論理学者ポストの1943年論文に負うと言っている（もっとも，ミスプリントで「1944年」となっている。そのミスプリントに言語学者が気づかないでいたらしいのは，言語学者の精神風土を暗示する事実かもしれない）。そしてポストこそは，ローゼンブルームの師であった。

組み合せ系の考えはポストの1936年論文に始まるらしい。この年にはまた，チューリングの論文も発表されて，後年彼を記念して「チューリング機械」とよばれる論理機械の提唱があったが，ポストの36年論文の内容はチューリングのそれに似ている。この辺，私は門外漢なのであまり詳しくは知らないが，さらにまた「マルコフ算法」も，多分，ほぼ同じころ，あの確率論で名高いマルコフの子息によって始められたようであるが，マルコフ算法も組み合せ系と大変よく似ている。こうして記号論理・形式体系・オートマトンの研究は，統一的な思考科学として実を結ぶに至る。フレーズ構造文法は，こういう流れの中で見ると，組み合せ系の一種だと思われてこよう。

電子計算機はオートマトンに現実の姿を与えたものだと言ってよかろうが，フレーズ構造文法の型の違いがある程度うまいぐあいにオートマトンの種類に対応することなど，言語と電子計算機との関係は単にアプリケーション例だということを越えて，結びつきの縁が深いものだと言えそうである。

こうしたあたり，もっと丁寧な叙述をしないと読者諸氏の納得が得られないだろうと，おそれはするが，何とも許された紙幅に比べ問題が大き過ぎるので，はしょらざるを得ない。先の文法や言語の定義にしても，自然言語での具体例がほしいとおっしゃる方もあろう。言訳めくが，紙幅を考えて，近ごろは日本でもこの種の論文がいろいろの学術誌に載ることだし，引例は断念して話の大筋だけは通そうと努めることにした。さて今，フレーズ構造文法の型に言及したから，最後にこのことについて述べておく。

6. 文法・言語の型

フレーズ構造文法，したがってフレーズ構造言語は，その作り出しの形式的性質によって，通常四つの型に分けられる。ただ分類できるというだけなら，さしたる価値もなかろうが，先まわりして言えば，次の定理が成り立つので型の別は大切である。

> **定理**　$i=1, 2, 3$ に対して，i 型言語ではない $(i-1)$ 型の言語がある；すなわち $(i-1)$ 型文法のうちには i 型文法では表わせないものがある。

そこで型の定義にとりかかろう。以下の述べ方は通説と見掛けが一部異なるが，そうしたほうがわかりよいと思ったので，あえて少々の全くノミナルな変更を加えた。

定義　一つの書き替え規則は，下記の条件 i を満たすとき，***i* 型**の形をしている。

0　規則は $\varphi\alpha\psi\to\varphi\beta\psi$ の型であり，α は一個の非末端記号，φ と ψ と β とは語彙に属する記号の連糸（空連糸でも可）である。

1　規則は $\varphi\alpha\psi\to\varphi\beta\psi$ の形であり，α と φ と ψ との条件は 0 と同じ，β は語彙に属する記号の少なくとも一個から成る連糸である。

2　規則は $\alpha\to\beta$ の形であり，α と β との条件は 1 と同じである。

3　規則は $\alpha\to\beta$ か $\alpha\to\beta\gamma$ かの形であり，α の条件は 2 と同じ，β は一個の末端記号，γ は一個の非末端記号である。

ここから，ただちに

定理　$i=1, 2, 3$ に対し，i 型の形をした規則は同時に $(i-1)$ 型の形をした規則でもある。

たとえば，先の「整数 → 0」や「整数 → − 正整数」は 3 型の形をした規則であり，それはまた 2 型，1 型，0 型の形をもしている。「正整数 → 正整数 数字」は 2 型の形をした規則であるとともに，1 型，0 型の形をもしている。たかだか 1 型，0 型である例は，前にあげたものの中にないので，改めて示せば（ただし，各記号がすべて V に属するとして）

たかだか 1 型：　A B → A c C

たかだか 0 型：　A B →

そこで

定義　文法 G の P の各規則が i 型の形をしていれば，その G は ***i* 型文法**である。その G によって生成される言語は ***i* 型言語**である。

たとえば P が六つの規則から成り，四つがたかだか 3 型で一つがたかだか 2 型で残りの一つがたかだか 1 型なら各規則は 1 型の形をしているから 1 型文法であり，また同時に 0 型文法でもある。

なお，型の名には番号以外の別名があるから，それを紹介しよう。

0 型：消しのある文脈感応/文脈依存（context-sensitive/context-dependent with erasing）

1 型：消しのない文脈感応/文脈依存（context-sensitive/context-dependent without erasing）

2 型：文脈無用（context-free）

3 型：有限状態（finite state），正則（regular），制約つき文脈無用（restricted context-free）

以下では，たかだか i 型であることを単に「i 型」といおう。0 型文法と 1 型以上の文法との違いは，記号を消し去る操作が許されるか否かにある。これが許されない1型以上の文法では，書き替えを行なう生成の過程でえられる連糸の長さが前の段階より短くなることはない。1 型以下の文法と 2 型以上の文法との違いは，矢印の左辺に，右辺で書き替えが認められている当の非末端記号一個以外の記号が現われる規則があるか否かである。そういう規則があれば，その左辺の当の非末端記号以外はそれに対する文脈的な条件を示すことになる。左辺が非末端記号ただ一個だけであれば，その前後がどんな連糸であっても，それにかかわりなく，当の非末端記号を右辺で指定するように書き替えてよい。このことが，文脈感応（文脈依存）とか文脈無用とかいう名のつけられたゆえんである。3 型文法と 2 型文法の違いは，改めて説くまでもあるまい。

さてそこで，有限状態オートマトンの働きは 3 型文法で記述できるということに注意しよう。五円か十円かの硬貨で二十円入れれば品物を出す自動販売器を考える。まさにちょうど二十円はいったことを確かめるアクセプタは次のように表わせる；五円硬貨・十円硬貨の投入をそれぞれ s，t で表わし，0 円，5 円，10円，15円，は

易しさ抜群の入門書！
基礎 情報理論
慶大教授　工博　藤田広一著　¥850
目次：情報理論の背景／情報の定量化／確率過程からの情報／エントロピー／通信路／符号化／連続確率関数の情報量／周波数スペクトラム／標本化定理／通信路の伝送特性／通信路の周波数特性と情報量／波形の伝送／波形の変換／信号の検出
アナログ電子計算機のプログラム　藤田広一　¥1,300
応用数学 —理工系コース—　山本格治　¥1,400
大学課程 微・積分学演習　西本勝之　¥1300, 1100
大学課程 微分方程式演習　西本勝之　¥1,300
自動制御概論　伊藤正美　¥1,300
線形電子回路　斎藤正男　¥1,100
東京都新宿区矢来町48
昭晃堂

いった状態をそれぞれS，A，B，Cで表わせば，

(1) $S \to sA$　　(2) $S \to tB$
(3) $A \to sB$　　(4) $A \to tC$
(5) $B \to sC$　　(6) $B \to t$
(7) $C \to s$

五円玉・十円玉・五円玉のこの順の投入で，ちょうど二十円になることは，上記の作り出しを使って

$S \Longrightarrow sA$　　(1)で書き替え
$\Longrightarrow stC$　　(4)で書き替え
$\Longrightarrow sts$　　(7)で書き替え

のようにして確かめられる。Sから出発し，上記の作り出しを使う範囲でえられる文は，*ssss*　*sst*　*sts*　*tss*　*tt*　の五種であり，そのほかにはない。これらが，実際に二十円になる入れ方と対応していることは，容易にわかると思う。もし十円玉・五円玉・十円玉という順で入れたとすれば，末端記号連糸は *tst* であるが，この連糸は今の文法の下での文ではない。なぜなら

$S \Longrightarrow tB$　　(2)で書き替え
$\Longrightarrow tsC$　　(5)で書き替え

で頭からの部分連糸 *ts* がえられるが，Cを書き替える作り出しは(7)しかなく，これを使ったのでは *tst* にはならないからである。

3型文法は有限状態オートマトンと対応するが，では2型文法は何と対応するか。それはプッシュダウン・オートマトンである。プッシュダウン・ストレージ（またの名スタッカ）は，プログラミング技法でよく使われるものである。多くのコンパイラ言語はこれを使って処理され，ソース・プログラムからオブジェクト・プログラムへのコンパイルが行なわれる。ALGOLを始めとしてコンパイラ言語は概して2型言語となるように設計されている。前にあげた「整数」の文法も2型文法であった。バッカス標準式は2型文法の記述法の一つである。そして2型文法は，自然言語を含めて「入れ子」(nesting)とよばれる言語構造を反映している（ついでながら，日本語の入れ子構造については，無論現代的な理論としてではないが，江戸時代の国学者の研究で，すでに知られていたと言ってよかろう）。

2型文法は扱いも割合に容易で，またその形式的性質についても相当よく研究されている。自然言語，とくに近代語の文法は，一般に，その中核的部分が2型文法でよく近似できる。算術式や命題論理式などの形式言語はそのまま，あるいはちょっと手を加えて，2型文法で規定できる。しかしながら自然言語の文法をとことんまでフレーズ構造文法の形で記述し切れるかといえば，それは1型や0型まで下げたとしても，まずは不可能である。この限界を突破する一方法としてチョムスキの変換文法が提案されたのであった。現在なお文脈感応文法や変換文法の形式的性質は十分には明らかになっていない。

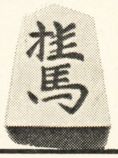
フレーズ構造文法の型について，次の注意をつけ加えておこう。α, β, γ を先に3型の定義で述べたのと同じ連糸だとして，ある文法GのPが　$\alpha \to \beta$　か　$\alpha \to \gamma\beta$　かの形の作り出しの集合であれば，先の定義に照らすと，このGは2型文法ということになる。しかし，L(G)とちょうど同じ言語を生成する文法G′の中には3型文法であるものが存在する。つまりこの場合には，作り出しをうまく作り変えてGと同等な3型文法にすることができる。だから，与えられたPの姿だけを見て，その文法が（2以下の）*i* 型であっても，たかだか *i* 型であると断定するわけにはいかないのである。

7. 終りに

本稿では話が抽象的レベルに終始した。この方面のことに関心のある読者にはすでに知っていることばかり述べ，なじみのなかった読者にはいささか不親切な述べ方になってしまったろうということを，反省している。さらに，これは初めのほうでお断わりしたが，取り上げた話題もわずかに二つである。ただ，私見では，数理言語学の領域を大別すれば計量言語学と算法言語学とになると思うので，それぞれから一例ずつを拾ったつもりである。ということは，たとえば算法言語学が本稿で扱ったような調子の文法研究の別名ではないということである。

本稿を読んでこの方面の研究に興味を覚えたという奇篤な方が，もしおいでだとすれば，その方は参考書を質問なさるかもしれない。しかし残念ながら，数理言語学全般にわたるような概説書はまだない。それほど，この学問は若いのである。主として日本語に関するこの方面の研究が載る雑誌には，計量国語学会（事務所：東京女子大学内）の『計量国語学』があることを，言い添えておく。

桂馬飛びパズル 2



先月号で筆者が提出された問題について多くの方から回答が寄せられていますが，3×4 の場合に16手より少ない回答はまだ見当りません。不可能であることの証明か，または可能な場合の実例が見つかりましたら，これからでも結構ですから編集部宛お知らせください。　　（編集部）

富士通(株)　池田敏雄

前号で紹介した桂馬飛びパズルの続きをお送りしよう。

桂馬飛びパズルを人間が考えるときは，人間の直観力やパターン認識能力を生かせるよう，若干問題を変形する必要がある。電子計算機で行なう場合には，桂馬飛びに本質があるのではなく，たとえば**図1**の3の場所から移動可能な場所は9，6および8であるから，**図2**のように3という場所を直線で9，6および8と結びつけた図にすれば，桂馬飛びは単に3から9あるいは3から6へ移動するのと同等である。したがって計算機で行なう場合には，各数字からの分岐と結合が本質となる。

人間の場合は，これをさらにきれいな図形（等価回路と名づけよう）に直すほうが考えやすい。それは**図3**のようになることは容易に確かめられるだろう。

こうすれば，3から5へ連続して桂馬飛びで行くことは3→9→7→2→5と移動することとなるから，人間が視覚によって思考するときには，きわめて都合がよい。

同様にして3×5，3×6* の場合の等価回路図も作ることができる。**図4**が3×5の等価回路であり，**図5**が3×6の等価回路となる。

まず，3×4の場合，A，B，C，Dがa, b, c, dに対応している位置に駒が配置されたときの解を示そう（**図6**，**7**）。

同図の右手に書いたような手順で A⇄a, …, D⇄d の入換えができる（16手）。

ついでに3×5（**図8**）および3×6（**図9**）の解も示しておこう。

次に，駒を任意の位置においてから出発して入換えを行なうとどうなるかを少し調べよう。

図10のような配置から出発するとすれば図のようになり，12手で入換えが可能である。したがって3×4の場合は，最初の駒の配置いかんでは，12手～16手のいずれかが可能となるらしい。というのは，17手を必要とする配置がまだ見つかっていないからである。そして図6が最悪（最も手数をよけいに必要とする）のパターンの一つであるらしい。読者の中で一つ「17手を必要とするパターンを見出す」か，あるいは「3×4の場合に16手以下で入換えが必ず行なえることの証明」ができたら，ぜひ教えていただきたい。

3×5の場合は，任意の配置から出発すれば，15手で入れ換えられる配置が最も多く，17手を必要とする配置はきわめて少ない。どうも18手を必要とするパターンは無いように思うのだが，それが存在するか(p. 56へ続く)

1	2	3	4
9	10	11	12
5	6	7	8

図1

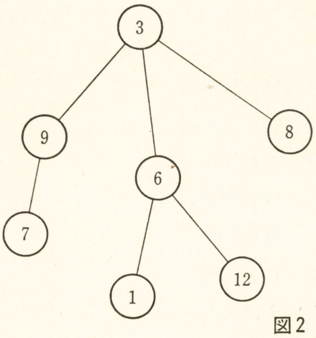


図2

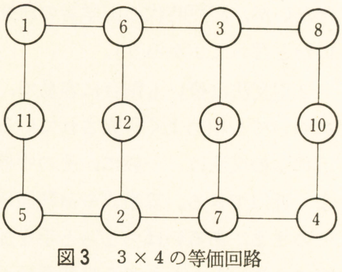


図3　3×4の等価回路

* 以下，たとえば，3行5列の桝目を簡単に3×5のように表わす。

1	2	3	4	5
11	12	13	14	15
6	7	8	9	10

(3×5)

1	2	3	4	5	6
13	14	15	16	17	18
7	8	9	10	11	12

(3×6)

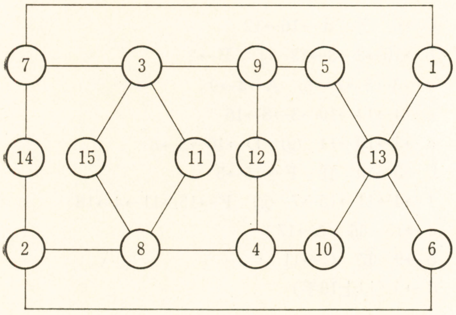


図4　3×5 の等価回路

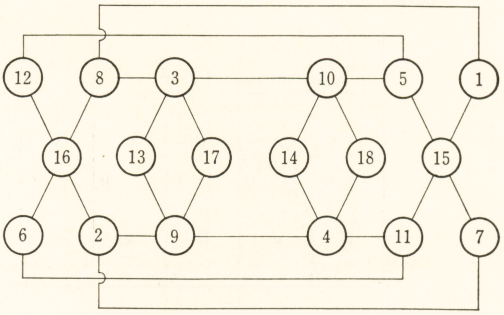


図5　3×6 の等価回路

A 1	B 2	C 3	D 4
9	10	11	12
a 5	b 6	c 7	d 8

図6

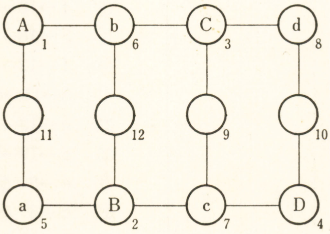


図7

① A→11　② D→10　③ c→4
④ B→7→9　⑤ b→12→2→7
⑥ a→2→12→6→1　⑦ b→2→5
⑧ B→7→2→12→6　⑨ C→9→7→2→12
⑩ c→7→2　⑪ d→3→9→7→4
⑫ D→8　⑬ c→7→9→3
⑭ C→2→7　⑮ b→2　⑯ A→5

図8

A 1	B 2	C 3	D 4	E 5
11	12	13	14	15
a 6	b 7	c 8	d 9	e 10

(3×5)

① B→14　② c→15　③ D→12
④ C→11→8→4　⑤ b→3→11→8→2
⑥ C→8　⑦ d→3→7
⑧ E→9→3　⑨ a→13→5→9
⑩ e→13→5　⑪ A→13→6
⑫ d→1→13→10→4
⑬ E→7→1→13→10
⑭ a→3→7→1
⑮ B→7　⑯ c→3
⑰ D→9（以上17手）

図9

A 1	B 2	C 3	D 4	E 5	F 6
13	14	15	16	17	18
a 7	b 8	c 9	d 10	e 11	f 12

(3×6)

① c→17　② D→18
③ B→9→13　④ e→4→14
⑤ F→11→15　⑥ f→16→6
⑦ E→12→16→2→9→4→11
⑧ d→5→12→16→2→9→4
⑨ C→10→5→12→16→2→9
⑩ F→5→12　⑪ b→16→2
⑫ a→15→5　⑬ A→15→7
⑭ a→15→1　⑮ B→3→8　⑯ e→10→5
⑰ c→3　⑱ D→10（以上18手）

図10

1	2	B 3	b 4
c 9	d 10	11	C 12
D 5	6	a 7	A 8

① D→11→1→6
② a→2→5→11→1
③ b→7→2→5→11
④ d→4→7→2→5
⑤ c→7→2 ⑥ A→10→4→7
⑦ B→8→10→4 ⑧ D→3→8→10
⑨ C→6→3→9 ⑩ a→6→3→8
⑪ b→1→6→3 ⑫ c→12（以上12手）

図11

1	b 2	3	4	D 5	6
13	A 14	E 15	C 16	a 17	f 18
F 7	d 8	B 9	e 10	c 11	12

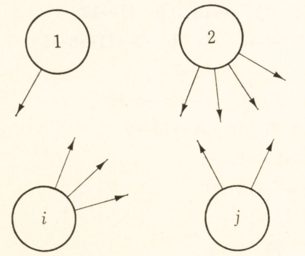
① C→6 ② d→16→12
③ b→16→8→3→13 ④ B→2
⑤ e→3→8→1 ⑥ A→4→9
⑦ c→4→14→10→3→8→16
⑧ a→3→10→14 ⑨ D→10→3→8
⑩ E→5→10 ⑪ F→15→5
⑫ f→4→11→15→7 ⑬ F→15→11→4→18
⑭ e→15 ⑮ A→17
⑯ b→9 ⑰ C→11
⑱ d→5（以上18手）

図12

A 1	B 2	C 3	D 4	E 5	F 6	G 7
15	16	17	18	19	20	21
a 8	b 9	c 10	d 11	e 12	f 13	g 14

① D→20 ② e→21
③ c→15 ④ B→10→4→12
⑤ f→18→2→10→4 ⑥ b→18→2
⑦ F→18→13 ⑧ B→6→18→9
⑨ f→12→6 ⑩ a→17→12
⑪ A→17→8 ⑫ a→17→1
⑬ g→19→10 ⑭ G→19→14
⑮ g→19→7 ⑯ E→17→12
⑰ e→5 ⑱ C→19→10
⑲ c→3 ⑳ d→16→4
㉑ D→11（以上21手）

図13



しないかの証明はできていない。

3×6の場合は，任意の位置から出発すれば，ほとんどの場合18手で入れ換えられる。**図11**に一例を示そう。

しかし，どうも19手を必要とする配置が存在するらしい。昔19手を必要とするパターンを二つ作ったことがあるのだが，残念ながらノートを紛失して再現できない。しかもそれが18手では入換え不可能を証明したものではなく，どうしても18手ではできなかったものであるだけに，19手を必要とする配置の存在を断言はできないのである。読者の中で19手入換えの配置ができたら教えてください。20手を必要とする配置は存在しないように思うのだがどうだろうか。

$3\times n(n\geqq 6)$の場合，第1行と第3行の対応する駒の入換えは$3n$手を必要とするのだが，$n\geqq 8$は解を得ることは容易であるので，ここでは$n=7$の場合の解のみを示しておこう（**図12**）。

このパズルは，さらに一般化すれば次のようになる。

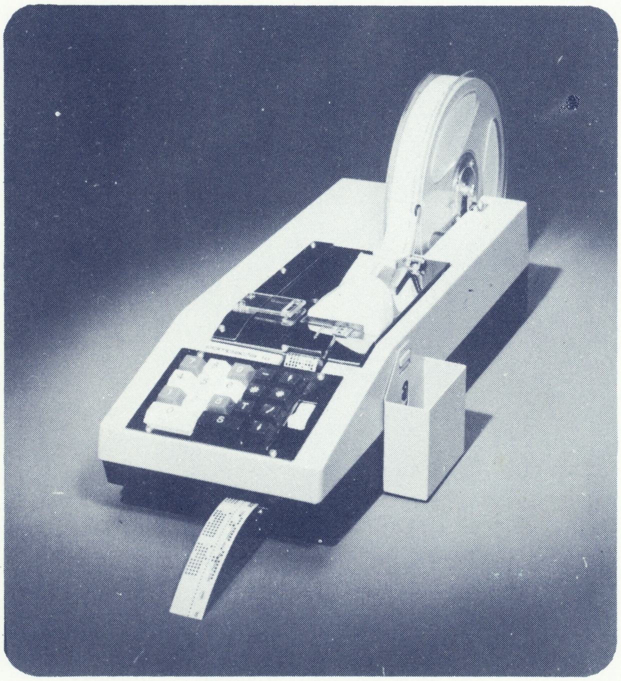
駒を置く場所を station とよび，それらを結合する線を結合線とよぶとき，n個の station とm本の結合線により図形ができる（**図13**）。その図で，$A_1, \cdots, A_l, a_1, \cdots, a_l$を任意に配置して，最小の手数で入換えを行なえ，という問題となる。

この種の問題は電子計算機で解を求めようとするときは，何か適当な方法を考えないと，ひじょうに多くの時間を必要とする。というより，高速計算機でもまったく不可能なものとなる。人間が行なう場合は，適当な直観力で解を求めるために，そう時間を必要としない。しかしさきに述べたように，3×6の場合，20手を必要とする配置が存在するかしないかという問題となると，証明が発見されないかぎり，電子計算機によって探す方法を作るほうが容易かもしれない。ここにも人間頭脳と電子計算機の共存の理由がありそうである。

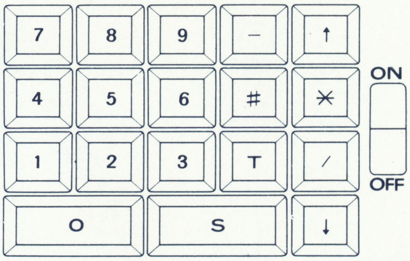
データパンチ処理に新鋭機

compacta 10

軽くて，コンパクト，誰にでも使えます



COMPACTA 10鍵盤配列



0～9…数字キー
—……マイナスキー
井……データーエンドキー
T……タブキー
S……ストップキー（レコードエンド）
↑……テープバックキー
⚹……デリーションキー
/……キャンセルキー
↓……テープフィードキー

現金正価 ¥270,000

compacta 10

- 軽量コンパクトです。
- 二重打鍵防止装置により重複鑽孔が防止されます。
- オートバック機構が付いていますから鍵盤のフォームポジションより手を離す必要がなく、テープバック・デリーションの操作は通常のオペレーションスピードで行えます
- 符号はプラグイン式万能マトリクスですから自由にコードを設定できます。
- 後部に入力コネクターが用意してありますからCOMPACTA·SERIESにすべて連動できます。

COMPACTA 10　仕様

- 打鍵鑽孔速度………1.500キャラクター/分50〜
  1.800キャラクター/分60〜
- 手動連続繰出速度…1.000キャラクター/分50〜
  1.200キャラクター/分60〜
- 単位……5·6·7·8単位紙テープ並にエッヂカード
- 駆動方式…マグネット・ドライヴ電源同期方式
- 寸法………………… 180(W)×450(L)×115(H)
- 重量…………………………………………約12kg
- 電源電圧…………………AC100±10V 50〜60〜
- 消費電力………………通常21W 打鍵時最大80W

compacta SERIES には compacta -20 加算装置及び compacta -40 タイプライター装置を連動したテープ鑽孔装置が用意されております。

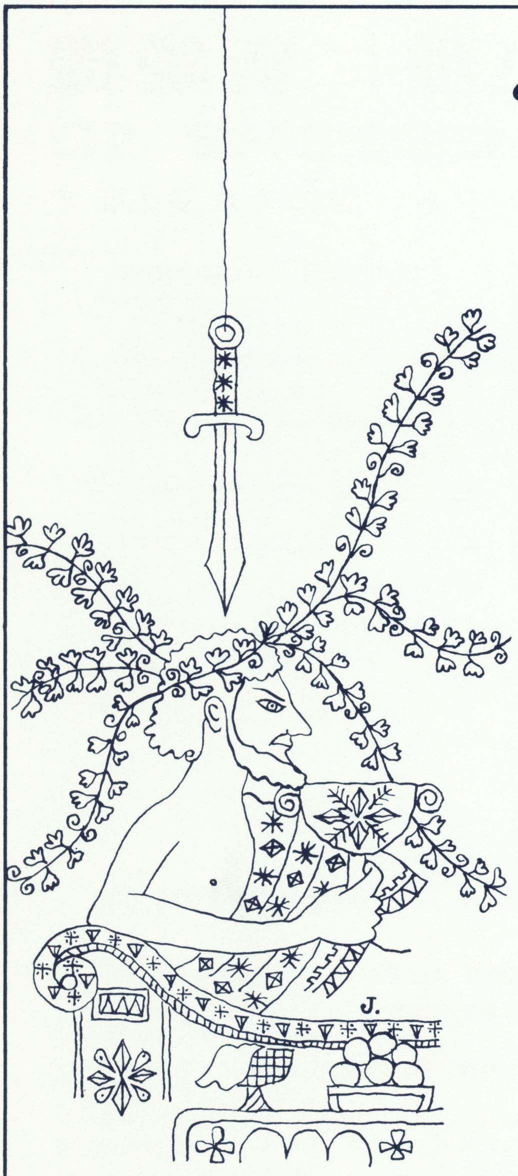
compacta 90VR ベリファイ装置でベリーが簡単にできます。

その他各種オプショナルユニットが用意されております。カタログ御請求下さい。

◉試用を御希望の方には貸出しも行っております。お気軽にお問合せ下さい。

製造元 東京データマシン株式会社
代理店 株式会社 日本計装

本社営業所　東京都港区南麻布2-13-19　高野ビル
TEL 03(453)2466 代表　郵便番号106
大阪営業所　大阪市西淀川区野里町1182　西淀ビル
TEL 06(471)7790, 7223　郵便番号555



“ダモクレスの剣”

＊紀元前４世紀のころ，シラクサの王ディオニュシオス１世は，専横その極に達していた。廷臣ダモクレスは，よく仕え，王の幸福を称賛していた。あるとき，王は盛大な饗宴をひらき，彼をねぎらった。ところが，１本の馬の毛につながれた鋭い剣が，彼の頭上につるされた。そして王は，おどろくダモクレスを前にして，支配者というものは，このように常に不安である，と説いたという。

＊この言い伝えから，“ダモクレスの剣”という言葉が生まれ，“危険な幸福”を象徴するようになった。

＊現代は，まさしく，この危険をはらんでいる。“ダモクレスの剣”をふり払うものは何か？　情報産業時代の中核とさえなったコンピュータを駆使するものこそ，その役割を担うものではなかろうか。日本ソフトウエアは，コンピュータの利用技術，すなわちソフトウエアの開発を目ざして，日夜たゆみなく前進しつづけている。もろ手をあげて，若々しいエネルギッシュな人材を，心から歓迎する。

こんな仕事をしています

★通産省の大型プロジェクトの推進
（超高性能電子計算機用ソフトウエアの開発）
★オンラインシステムの開発
★日本万国博運営の情報システムの開発
★学園都市の設計・運営システムの開発
★予算管理システム（ＰＰＢＳ）の開発

● 入社ご希望の方は，管理部人事課宛
履歴書をお送り下さい。

日本ソフトウエア
東京都港区芝西久保明舟町11第11森ビル
電話（591）8241（代）・郵便番号105

連載　PL／I⑤

関数とサブルーチン

久保　宏志

PL/Iにおけるサブルーチンの意味を知り，具体例によって機能・手続きおよびその実行などについて調べてみよう

1.　サブルーチンとは

ふつうには，三角関数，対数関数，平方根の計算などのような，とくに使用頻度が高くかつ比較的単純な機能をはたす独立したプログラムを**サブルーチン**とよんでいるかもしれないが*，ここでは，ある一定の機能をはたすプログラムが次の二条件をみたすものであるとき，それをサブルーチンとよぼう。第一に，このプログラムを働かせるには，呼び出すという手順をふまなければならないこと。第二には，このプログラムは呼び出すための名前をもつものであること。この定義はまえの定義に比べるとずっと広いものになっているが，機能に名前を与えその名前によって機能を引用するという考え方がサブルーチンの最も本質的なところであり，サブルーチンの活用が有効であるのはこの本質によるのである。

サブルーチンをこのように規定すると，ある問題を解くためのプログラムは一個のサブルーチンであるとみることができる。このサブルーチンのはたすべき機能が単純であるか，あるいは，たとえ複雑であってもプログラム言語がその問題向きにできていて小さいプログラムにまとまるときには，サブルーチンの機能をいくつかのより単純な機能に分解する必要はないかもしれない。しかし，機能が複雑になってくると事情は変わってくる。われわれは，複雑な機能を実現する手順を組み立てるとき，その機能をより単純な機能に分解し，その各機能を実現する手順は**ブラック・ボックス**のまま全体の手順を組み立てるということをする。こうすることが全体の見通しをよくし，思考の経済上も望ましいからである。人間の思考様式がこうであり，一般に言語が思考の表現手段であるなら，プログラム言語による思考の表現としてのプログラムもまたこの思考様式を直接的に反映するものであるべきだろう。各機能単位にサブルーチンを対応させサブルーチンの系によって全体の論理を構成するという態度が，プログラミングに際しての最も基本的な態度として要求されるのである。この点においてサブルーチンという概念が重要な意味をもってくるのである。

Hiroshi Kubo
富士通（株）

* “単語帖”，本誌Vol. 1，No. 2，p. 56（4月号，1969）参照

2.　デザルグの定理

サブルーチンの定義の仕方，それの呼び出し方を説明するまえに，プログラム例と実行結果を示そう（図1，2）

射影幾何学にデザルグ（Desargues）の定理というものがある。それは，同一平面上の二つの三角形 ABC, XYZ があるとき，三直線 AX, BY, CZ が一点に会するという命題（命題1）と二つの三角形の対応する辺AB と XY, BC と YZ, CA と ZX の交点が一直線にあるという命題（命題2）が同値であるという（図3）。

プログラム例はこの二つの命題の意味を解析幾何学的に表現するものである。考え方を簡単に述べよう。

点と線を三つ組 (α, β, γ) で表わそう。すなわち点の表現 (α, β, γ) は座標 $(\alpha/\gamma, \beta/\gamma)$ をもつ点を意味し，線の表現 (α, β, γ) は直線 $\alpha x+\beta y+\gamma=0$ を意味するものとする。このとき，二点A : (a_1, a_2, a_3)，B: (b_1, b_2, b_3) を通る直線は

$$(a_2\cdot b_3-a_3\cdot b_2,\ a_3\cdot b_1-a_1\cdot b_3,\ a_1\cdot b_2-a_2\cdot b_1)$$

と表わすことができる。これを line(A, B) で示そう。二直線A, Bの交点もこれと同じ三つ組で表わすことができる。これを point (A, B) で示そう。また，点Aが線B上にあること，あるいは，点Bが線A上にあることは

$$a_1\cdot b_1+a_2\cdot b_2+a_3\cdot b_3=0$$

と表わすことができる。この式を on (A, B) で示そう。

図1

```
LEVEL                                 FACOM230-50 PL/1 COMPILER

SOURCE LISTING

   1    DESARGUE:PROCEDURE OPTIONS(MAIN) ;                                               BIT02010
   2             DECLARE    (A,B,C,X ,Y ,Z )(3) CHARACTER(2) VARYING;                   BIT02020
   3                        A(1)='A1';      A(2)='A2' ;                                  BIT02030
   5                        B(1)='B1';      B(2)='B2' ;                                  BIT02040
   7                        C(1)='C1';      C(2)='C2' ;                                  BIT02050
   9                        X(1)='X1';      X(2)='X2';                                   BIT02060
  11                        Y(1)='Y1';      Y(2)='Y2';                                   BIT02070
  13                        Z(1)='Z1';      Z(2)='Z2';                                   BIT02080
  15                        A(3),B(3),C(3),X(3),Y(3),Z(3)='1';                          BIT02090
  16             DECLARE    (WK1,WK2,WK3,WK4)(3) CHARACTER(256) VARYING;                BIT02150
  17             LINE :                                                                  BIT02180
                 POINT: PROCEDURE ( P, Q, R ) ;                                          BIT02190
  18                    DECLARE    (P,Q,R)(*) CHARACTER(*) VARYING ;                     BIT02200
  19                    EXPR: PROCEDURE( S,T,U,V,W);                                     BIT02220
  20                          DECLARE   ( S,T,U,V,W) CHARACTER(*) VARYING;               BIT02230
  21                          PUT STRING(S) EDIT                                         BIT02235
                                        ( '(',T,'*',U,'-',V,'*',W,')' )(A);BIT02240
  22                          END EXPR ;                                                 BIT02250
  23                    CALL EXPR( P(1), Q(2),R(3), Q(3),R(2) );                         BIT02260
  24                    CALL EXPR( P(2), Q(3),R(1), Q(1),R(3) );                         BIT02270
  25                    CALL EXPR( P(3), Q(1),R(2), Q(2),R(1) );                         BIT02280
  26                    RETURN ;                                                         BIT02290
  27             ON:    ENTRY  ( P,Q ) ;                                                 BIT02320
  28                    PUT EDIT(      P(1),'*',Q(1),                                    BIT02330
                                 '+',P(2),'*',Q(2),                                      BIT02340
                                 '+',P(3),'*',Q(3),'=0' ) (A) ;                          BIT02350
  29                    END LINE ;                                                       BIT02360
  30      /*  FIRST PREDICATE */                                                         BIT02390
                 CALL  LINE( WK1, A, X  );  /* WK1=THE LINE THRU A AND X  */             BIT02400
  31             CALL  LINE( WK2, B, Y  );  /* WK2=THE LINE THRU B AND Y  */             BIT02410
  32             CALL  LINE( WK3, C, Z  );  /* WK3=THE LINE THRU C AND Z  */             BIT02420
  33             CALL  POINT( WK4, WK1, WK2 );                                           BIT02430
  34                                        /* WK4=THE INTERSECTION                      BIT02440
                                                   OF A_X AND B_Y */                     BIT02450
                 PUT PAGE LINE(5);        CALL ON( WK4, WK3 );                           BIT02460
  36                                       /*THE POINT  A_X & B_Y                        BIT02470
                                                   IS ON THE LINE C_Z  */                BIT02480
        /*  SECOND PREDICATE */                                                          BIT02490
                 CALL  LINE( WK3, A, B  );  /*  A_B */                                   BIT02500
  37             CALL  LINE( WK4, X, Y  );  /*  X_Y */                                   BIT02510
  38             CALL  POINT ( WK1, WK3, WK4) ;  /* A_B & X_Y */                          BIT02520
  39             CALL  LINE( WK3, B, C  );  /*  B_C */                                   BIT02530
  40             CALL  LINE( WK4, Y, Z  );  /*  Y_Z */                                   BIT02540
  41             CALL  POINT ( WK2, WK3, WK4) ;  /* B_C & Y_Z */                          BIT02550
  42             CALL  LINE( WK3, WK1, WK2) ; /*(A_B & X_Y)_(B_C & Y_Z) */               BIT02560
  43             CALL  LINE( WK1, C, A  );  /*  C_A */                                   BIT02570
  44             CALL  LINE( WK2, Z, X  );  /*  Z_X */                                   BIT02580
  45             CALL  POINT( WK4, WK1,WK2) ;     /* C_A & Z_X */                        BIT02590
  46             PUT SKIP(3);             CALL ON( WK4, WK3 );                           BIT02600
  48                         /*  THE POINT C_A & Z_X                                     BIT02610
                                  IS ON THE LINE (A_B & X_Y)_(B_C & Y_Z) */              BIT02620
                 END  DESARGUE ;                                                         BIT02630
```

二つの命題の意味は，line，point，onを使って次のように表わすことができる。まず，命題1は

on (point(line(A, X), line(B, Y)), line(C, Z))

と表わすことができる。そして，命題2は

on(line(point(line(A, B), line(X, Y)),
　　point(line(B, C), line(Y, Z))),
　point(line(C, A), line(Z, X)))

と表わすことができる。

プログラムもまたこのように書かれているとわかりやすいのだが，プログラム例はこの表現そのものではなくて，これをくずしたものになっている。つまり，点や線を表わすのにpoint(A, B)やline(A, B)をそのまま使わないで，これで表わされる点や線に名前をつけることをやって，かっこのない形にくずしてある。こうすることがよりよいからではなくて，PL/Iではこうせざるをえないからである。その理由については後述する。

このプログラムは前回までの説明ではふれなかったような書き方をいくつか含んでいる。このうち，今回のテーマに直接には関係しないものについて説明しておこう。

第15文

A(3), B(3), C(3), X(3), Y(3), Z(3) = '1';

は，代入記号の右辺の値を左辺のすべての変数に左から順に代入するという意味をもつ。

FACOM 230-50 PL/I は文字データの最大長を 256

```
((1*X1-A1*1)*(B1*Y2-B2*Y1)-(A1*X2-A2*X1)*(1*Y1-B1*1))*(C2*1-1*Z2)+((A1*X2-A2*X1)*(B2*1-1*Y2)-(A2*1-1*X2)*(B1*Y2-B2*Y1))*(1*Z1-C1*1)+((A
2*1-1*X2)*(1*Y1-B1*1)-(1*X1-A1*1)*(B2*1-1*Y2))*(C1*Z2-C2*Z1)=0

((1*A1-C1*1)*(Z1*X2-Z2*X1)-(C1*A2-C2*A1)*(1*X1-Z1*1))*(((A1*B2-A2*B1)*(X2*1-1*Y2)-(A2*1-1*B2)*(X1*Y2-X2*Y1))*((B2*1-1*C2)*(1*Z1-Y1*1)-(
1*C1-B1*1)*(Y2*1-1*Z2))-((A2*1-1*B2)*(1*Y1-X1*1)-(1*B1-A1*1)*(X2*1-1*Y2))*((B1*C2-B2*C1)*(Y2*1-1*Z2)-(B2*1-1*C2)*(Y1*Z2-Y2*Z1)))+((C1*A
2-C2*A1)*(Z2*1-1*X2)-(C2*1-1*A2)*(Z1*X2-Z2*X1))*(((A2*1-1*B2)*(1*Y1-X1*1)-(1*B1-A1*1)*(X2*1-1*Y2))*((1*C1-B1*1)*(Y1*Z2-Y2*Z1)-(B1*C2-B2
*C1)*(1*Z1-Y1*1))-((1*B1-A1*1)*(X1*Y2-X2*Y1)-(A1*B2-A2*B1)*(1*Y1-X1*1))*((B2*1-1*C2)*(1*Z1-Y1*1)-(1*C1-B1*1)*(Y2*1-1*Z2)))+((C2*1-1*A2)
*(1*X1-Z1*1)-(1*A1-C1*1)*(Z2*1-1*X2))*(((1*B1-A1*1)*(X1*Y2-X2*Y1)-(A1*B2-A2*B1)*(1*Y1-X1*1))*((B1*C2-B2*C1)*(Y2*1-1*Z2)-(B2*1-1*C2)*(Y1
*Z2-Y2*Z1))-((A1*B2-A2*B1)*(X2*1-1*Y2)-(A2*1-1*B2)*(X1*Y2-X2*Y1))*((1*C1-B1*1)*(Y1*Z2-Y2*Z1)-(B1*C2-B2*C1)*(1*Z1-Y1*1)))=0
```

図 2

字に制限しているが，第21文の

PUT STRING(S) EDIT
('(', T, '*', U, '−', V,
'*', W, ')')(A);

という書き方は，連結演算を使って

S='('||T||'*'||U||'−'
||V||'*'||W||')';

と書いたのでは演算の中間結果の最大長が256字を越えて文法違反を犯すことになるので，それを避ける目的で利用したのである。ふつう，PUT文による出力先がファイル名で指定されるデータ・セットであるのに対し

PUT STRING (単元文字変数) ……

という形の **STRING オプションをもつ PUT 文**では，出力先が単元文字変数であり，そのなかに左端から順に文字の流れを組み立てることになる。文字流の組み立て方は非 PRINT ファイルに対するときと同じである。

文字データの最大長についての制限が充分大きいところに設定されているような言語なら LINE: から END LINE; までを**図4**のように書きかえることもできる。

3. 手続きの定義

1. 入口点と入口名

図1のプログラムは

DESARGUE : PROCEDURE
OPTIONS(MAIN);

で始まり

END DESARGUE;

で終わっている。これは，PROCEDURE 文から END 文までのプログラムが定義する機能に DESARGUE という名前を与えているのである。

PROCEDURE 文から END 文

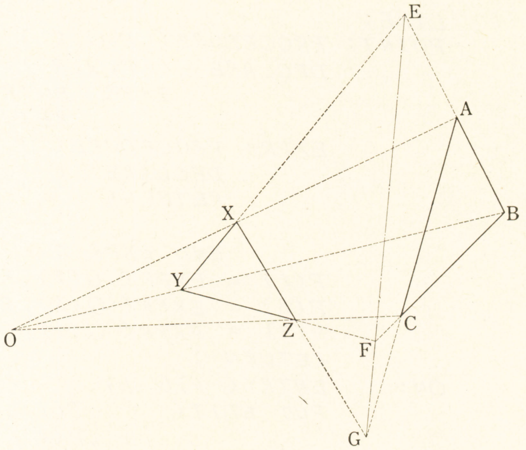


図 3

までを**手続き** (procedure) とよぶが，一般に，機能に名前を与えるには，その機能を手続きの形で定義するのである。これが PL/I におけるサブルーチン定義の基本的考え方である。

このように，PL/I ではサブルーチンは手続きの形をとるが，機能と手続きの対応は一対一である必要はない。2個以上の機能を一手続きの形で定義することもできる。それには，第二の機能を定義する文の先頭に**ENTRY 文**をおき，その全体がある手続きに含まれるようにすればよい。PROCEDURE 文には，手続きの範囲を定めるというはたらきと，機能を定義する文の先頭を定めるというはたらきがあるが，ENTRY 文はこの後者のはたらきだけをもつ。PROCEDURE 文，ENTRY 文にたてられたラベルが，定義された機能の名前となる。サブルーチンを定義するとは，機能を定義しそれに名前を与えることであるから，これらの文のラベルを省略することは許されない。すくなくとも1個のラベルは必要である。2個以上のラベルがあれば，1個の機能に2個以上の名前を与えることになる。これらのラベルの書き方は一般の文のラベルと類似しているけれども，その意味はまったく異なる。単に文そのものに名前をつけるのと，一連の文によって意味される機能に名前をつけるのとの違いである。そこでこれらを区別して，PROCEDURE 文と ENTRY文のラベルのことを，特に**入口名** (entry name) という。これらの文が手続きへの**入口点** (entry point) を定めるということからきている。

図1のプログラムの第17文

LINE : POINT : PROCEDURE(P, Q, R);

から第29文

END LINE;

```
LINE:
POINT: PROCEDURE( P,Q,R ) ;
       DECLARE  ( P,Q,R )(*) CHARACTER(*) VARYING,
                EXPR  ENTRY
                      RETURNS( CHARACTER(256) VARYING) ;
       EXPR: PROCEDURE( T,U,V,W ) CHARACTER(256) VARYING ;
             DECLARE  ( T,U,V,W ) CHARACTER(*);
             RETURN(  '(' || T || '*' || U ||
                      '-' || V || '*' || W || ')' );
             END EXPR ;
       P(1) = EXPR( Q(2),R(3), Q(3),R(2) );
       P(2) = EXPR( Q(3),R(1), Q(1),R(3) );
       P(3) = EXPR( Q(1),R(2), Q(2),R(1) );
       RETURN ;
ON:    ENTRY( P,Q );
       PUT EDIT(      P(1),'*',Q(1),
                  '+',P(2),'*',Q(2),
                  '+',P(3),'*',Q(3),'=0' ) (A) ;
       END LINE ;
```

図 4

までは一つの手続きである。この手続きは，第17文と第27文

ON：　ENTRY(P, Q);

によって二つの機能を定義し，一方の機能に LINE と POINTの二つの名前を与え，もう一方の機能にONという名前を与えている。LINE, POINT, ON は入口名である。

2.　情報伝達の方法

機能に名前をつけるにはどうすればよいかを知ったが，機能を定義することに話を進めよう。ここで問題になるのは，手続きの機能を引用する側からの入力をどのように受け入れ，出力をどのように返すかということである。その方法について述べよう。

第一は，変数名を共通にするという方法である。すなわち手続きとその機能を引用する側との間にいくつかの共通変数を取り決め，それによって情報伝達をするというものである。とくに説明するまでもないだろう。

入力データを与える変数や結果をもらう変数が機能を引用するたびに変わるような場合にもこの方法を適用しようとすると，そのたびにデータを共通変数に移したり，あるいはそこから移したりしなければならない。このような煩わしさを避けるために，共通の変数をもつのではなく，情報伝達に関係する変数の名前が機能の引用のたびに変わっても，手続きの側ではそれらをつねにおなじ名前で引用するという方法がある。たとえば図1のプログラムの

LINE：POINT：PROCEDURE(P, Q, R);

は，手続きの機能が LINE あるいは POINT で，引用されるときの3個のインタフェース情報を，手続きのなかではP, Q, R という名前でよぶことを意味している。P, Q, Rによるこの名前がえは手続きの機能が引用されるたびになされるのであるから，そのたびに引用側から異なる変数が与えられようとも，手続きのなかではつねに同じ名前で引用することができるのである。

このP, Q, Rのことを**パラメタ** (parameter) という。パラメタに対応する実体は機能が引用されたときはじめて確定するが，パラメタが何に対応しうるかは規定しておく必要がある。この規定を行なわなくてもよいとする言語もあるけれども(たとえばALGOL)，PL/Iではパラメタ引用の意味が静的に決まるように，対応実体の性質をパラメタの属性として宣言しておかなければならない。図1のパラメタの宣言

DECLARE (P, Q, R) (*) CHARACTER(*)
VARYING;

は，P, Q, R がVARYING CHARACTER の一次元配列に対応しうるものであることを示す。＊は長さと上下限は任意であることを示す（このプログラム例では

DECLARE(P, Q, R) (3)CHARACTER(＊)
VARYING;

とするのが適切であるが，FACOM230-50 PL/Iは一つの宣言のなかに*と*以外の指定を混在させることを認めない)。**長さや上下限の＊による指定**がこのような意味をもつのはパラメタだけであり，対応させるべき実体にある程度の自由度を与える効果をもつ。これを使えば，たとえば，任意の二つの行列の行列積A＝B･Cを計算する手続きは**図5**のように書くことができる。LBOUND(B, 1)は配列Bの第一次元下限を表わし，UBOUND(B, 1) はBの第一次元上限を表わす。＊によって任意の上下限を

```
PRODUCT:  PROCEDURE( A,B,C ) ;
          DECLARE ( A,B,C )(*,*) REAL FLOAT DECIMAL,
                  ( I,J,K ) REAL FIXED BINARY ;
          DO I= LBOUND(B,1) TO UBOUND(B,1) ;
             DO J= LBOUND(C,2) TO UBOUND(C,2) ;
                A(I,J) = 0 ;
                DO K= LBOUND(B,2) TO UBOUND(B,2) ;
                A(I,J) = A(I,J) + B(I,K)*C(K,J)  ;
                END ;
             END ;
          END ;
          END PRODUCT ;
```

図 5

もつ配列に対応させることができるが，その配列要素はREAL FLOAT DECIMAL データでなければならず，COMPLEX データや FIXED データであることではできない。この点が重要なところである。

共通変数によってもパラメタによっても手続きから計算値を返すことはできるが，計算値を返すための特別の方法がある。この方法では手続きのなかで

RETURN(式);

のように**式をともなうRETURN文**をもちいる。こうすると式の計算値が返されることになるのである。図4のプログラムにはこのようなRETURN文の例がある。

このようなRETURN文を書くだけでは十分でなく，**計算値の属性**，つまり，計算値をどのような属性で表現するかも宣言しておかなければならない。その仕方は図4のPROCEDURE文が示している。ここにはENTRY文で宣言する場合は含まれていないが同様にすればよい。一般に，1個の手続きはこのようなRETURN文を複数個含み，そのいずれの計算値が返されるかをプログラムの静的構造だけからは知ることができないが，どの計算値が返される場合でも，値は同じ属性で表現されることになる。一手続きが複数個の機能を含むとき，それらが与える計算値の属性は一致しなくてよい。ということは，PROCEDURE 文で宣言しても，その宣言がENTRY文にも適用されることはなく，PROCEDURE 文だけでなく，すべての ENTRY 文で宣言しなければならないということでもある。宣言が省略されたときには，算術データ属性の省略時解釈が適用される。

なお，この方法で返すことができるのは1個の単元データだけであり，計算値が配列の形をとるような場合には利用できない。

3. 定義の完結

入口点に始まった機能の定義はRETURN文で完結する。したがって，RETURN(式);で計算値を返すのは最後に行なわなければならない。また，手続きの END 文は手続きの範囲を定めるはたらきをもつほかに，RETURN文の機能をもつ。図1のプログラムにおいて，LINEとPOINTに対する機能の定義はRETURN文で完結しているが，ONに対してはEND文で完結している。手続きがとくに出発手続きである場合には，RETURN 文はSTOP文の意味をもかねる。したがって，図1のDESARGUE手続きのように，RETURN文もSTOP文も使わずにEND文で実行を終了させることができるのである。

4. 手続きの実行

1. サブルーチンとして

機能を定義し，それに名前を与える方法を知ったが，

日本記録紙のフォーム印刷

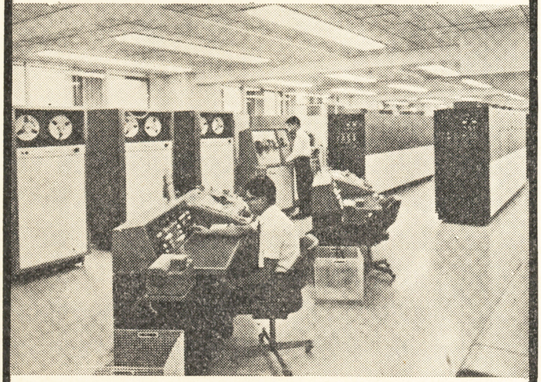
さん孔テープ・テレタイプ受信紙

製造販売

アコーダービジネスフォーム(株)

東京都中央区日本橋茅場町3－9第二広田ビル
TEL東京(03)667－7271(代)

コンピューター技術
電算マン完全教育！
情報革命・電算機時代に備へよ！



●7月生募集！ 毎日受付中 <学則 200円>
通学の部と通信教育部
《科目》プログラマー
オペレーター
システムエンジニアー
■本科六ケ月、高等科六ケ月、
昼間部、夜間部、日曜の部、通信教育部
■男女年令不問、学歴高卒程度
■各科定員100名 （就職率100%）
■電子計算士号授与（先着順定員〆切）
学長 高世一成博士
権威ある本邦唯一の全国的組織をもつ
日本電子計算学院
東京本校 東京都新宿区新宿1－86（〒160）（地下鉄新宿御苑前） TEL＝03(354)1756～8
横浜校 横浜市中区扇町3－8－1（〒232） TEL＝045(681)2696
静岡校 静岡市本通6－1（〒420）（本通交番前） TEL＝0542(52)7720
名古屋校 名古屋市千種区東山通5－19（〒464）（地下鉄東山公園前） TEL＝052(781)0508
大阪校 大阪市浪速区新川2－69（〒556）（地下鉄ナンバ、大阪球場前） TEL＝06(641)3841
広島校 広島市大須賀町13－9（〒730）（広島駅前中央郵便局西側） TEL＝0822(62)8782
北九州校 北九州市小倉区京町5丁目179（〒802） TEL＝093(52)3431～5番（小倉駅前）
福岡校 福岡市明治町2－21（〒812）（博多駅前） TEL＝092(43)4822
札幌校 札幌市北二條東1－2（〒063）（創成川西向） TEL＝0122(25)1428

次にその機能を引用する方法について述べよう。その第一の方法は**CALL文**による方法である。たとえば，図1のプログラム中の文

```
CALL LINE(WK1, A, X);
```

が示すように，CALL 文は入口名とそれに続くパラメタに対応させるべきもののリストによって機能を引用する。このように，CALL 文によって機能を引用することを**サブルーチンとして引用する** (subroutine reference) という。この引用によっては RETURN (式)；で返される結果を受け取ることはできない。

機能の引用がなされたとき手続きの実行は始まるが，このことを**機能の引用によって手続きを呼び出す** (invoke) という。

WK1, A, X を**アーギュメント** (argument) とよぶ。アーギュメントとパラメタの対応はそれらのリスト中での位置によって，1番目のアーギュメントは1番目のパラメタに，2番目のアーギュメントは2番目のパラメタにというようにつけられる。アーギュメントとパラメタの個数は一致しなければならない。個数の一致だけでなく，個々のアーギュメントは対応パラメタの宣言と矛盾するものであってはならない。定数や式もアーギュメントとすることができ，そのときには定数や式そのものがパラメタに対応するのではなく，中間的な変数が定義され，それに定数や式が定めるデータが代入されその中間的な変数がパラメタに対応する。この中間的な変数を**仮アーギュメント** (dummy argument) とよぶが，それは式の構文と各項の属性から定まる属性をもつので注意を要する。これが CHARACTER 属性や BIT 属性をもつ場合には，パラメタ宣言の長さ指定を * とすることで矛盾を避けることができるが，算術データの属性をもつときにはそのような便利な手はない。一般的にいって式が算術式であるときに，式の属性とパラメタの属性が一致することはありえないことである。細心の注意を払って一致させえたとしても，ちょっとしたプログラムの手直しでたちまちのうちにくずれてしまう。

たとえば

```
SUBR : PROCEDURE(X);
        DECLARE X FIXED DECIMAL(5,2);
```

とあるとき

```
CALL SUBR(0);
```

とするのは誤りである。0はDECIMAL FIXED (1,0) という属性をもちXの宣言とは一致しないからである。この誤りを避ける一つの手は，定数とか式とかをアーギュメントとすることをやめ，必ずその値を計算して変数に

代入し，その変数をアーギュメントとすることである。もう一つは，このことを暗黙のうちに行なわせるために，**ENTRY属性**を使って，

```
DECLARE SUBR ENTRY (FIXED DECIMAL
(5,2));
```

のようにパラメタの性質を引用の側で宣言しておくのである（同じことをなぜ二度も書かなければならないかについては次回で述べる）。こうしておくと，パラメタ宣言に矛盾しないものが手続きに渡される。すなわち，パラメタ宣言に矛盾しない変数アーギュメントに対しては変数名そのものが渡され，それ以外の変数アーギュメントや定数や式に対してはパラメタが要求している性質の仮アーギュメントがつくられ，それに値が代入されて，その仮アーギュメント名が渡されるのである。

2. 関数として

手続きの機能を引用するもう一つの方法は，CALL文によらず，図4のプログラム中の文

```
P(1)=EXPR(Q(2),R(3),Q(3),R(2));
```

が示すように，式中で**関数として引用する**（function reference）のである。この引用によってのみ RETURN（式）；で返される値を受けとることができ，この値が式の計算に使われるのである。したがって，機能の一部として RETURN（式）；が含まれていないときには，この引用の仕方は誤りである。

この引用をするためには入口名に **RETURNS 属性**を与えて関数値の属性を宣言しておかなければならない。その仕方は図4のプログラムが示している。RETURN文のためと引用のために同じことを二度宣言しなければならないことになるが，これはパラメタの意味を定義するためと仮アーギュメントをつくるために同じ宣言をしなければならないのと同じ理由による。

5. 総称入口

以上述べてきたように，ある入口名による機能の引用では，アーギュメントの個数は固定している。引用のたびにアーギュメントの個数を変えるようなことはできない。手続きに渡されるアーギュメントの属性も一通りである。実数アーギュメントとしたり複素数アーギュメントとしたりして機能を変えるようなことはできない。関数値の属性も一通りである。

このような制約を越えるために**総称入口名**（generic entry name）を定義することができる。いくつかの入口名をあつめて一つのファミリーをつくり，それを総称する名前を与えるのである。たとえば，ALPHA と BETA という二つの入口名が

```
DECLARE ALPHA ENTRY (REAL FLOAT
        DECIMAL),
        BETA ENTRY(COMPLEX FLOAT
        DECIMAL);
```

と宣言されているとき，**GENERIC属性**を使って

```
DECLARE GAMMA
   GENERIC (ALPHA ENTRY(REAL FLOAT
        DECIMAL), BETA ENTRY
        (COMPLEX FLOAT DECIMAL))
```

とすると，GAMMAはALPHAとBETAを総称する入口名となる。

二つの変数が

```
DECLARE (X REAL, Y COMPLEX) FLOAT
        DECIMAL;
```

と宣言されているとしよう。このとき

```
CALL GAMMA(X);
CALL GAMMA(Y);
```

は，それぞれ

```
CALL ALPHA(X);
CALL BETA(Y);
```

を意味するのである。

PL/I にはその機能が言語仕様の一部として定義されているような手続きがかなりたくさんある。それらは**常設関数**（built-in function）とよばれ，関数として引用することができるが，大部分のものが総称的である。変数Xの値が実数xであればABS(X)は$|x|$を関数値とし，Xの値が複素数$x+iy$であれば$\sqrt{x^2+y^2}$を関数値とするといった具合である。

おもな常設関数については解説するつもりであるが，その全体を紹介する余裕はないので PL/I の文法書を参照されたい。

次回は「ブロック構造」の予定

コンピュータ・サイエンス誌
bit
ファイルができました
bit が半年分とじこめる
丈夫で美しいファイルです。
バックナンバーの
整理・保存にぜひご利用下さい。
定価は 200 円です。ご注文は最寄の書店でお申し込み下さい。
共立出版株式会社

日本評論社

電子計算機のプログラミング

一松 信著　一五〇〇円

抽象化した架空の機械MS-2を設定し、一般的なプログラミング技法がアセンブラ言語で解説されている。実用のための数値計算のプログラムより、むしろシステムプログラムの基本技法を中心にして、これと関連して一つの問題にもいろいろのプログラムが可能であることやいろいろな方式があることを示しそのため分岐やループやサブルーチン・リンクなどの制御に関する話題について詳しく解説している。電子計算機体系の勝負どころがソフトウェアの面にあるといわれている今日、いわばプログラム論の基礎資料として本書の果す役割は大きい。

電子計算機時代（経営に対する可能性）

G・バーク＝フォーチュン編／一松 信訳　五四〇円

電子計算機の経営への応用ないしは電算機産業の将来について論じた斬新かつ高水準の入門書。

電子計算機と二進法

一松 信著　八〇〇円

電子計算機の回路設計などへの応用を念頭におきながら、二進法について解説したものである。

FORTRAN自習書

スミス＝ジョンソン著／田中穣二・根岸竜雄訳　七〇〇円

FORTRANの21個の命令を使って、プログラムの作り方を自学自習できるよう平易にとく。

東京都新宿区須賀町／振替東京16番

コンピュータ・ニュース

☆えがかれた磁気地図

通産省地質調査所では，日本周辺の大陸ダナの資源分布を知るために空中磁気探査法で集めたデータの処理にコンピュータを使い，ディジタル・プロッタによって磁気地図を描かせた。空中磁気探査法とは，飛行機に積んだ磁力計とドプラー・カメラで空中磁力の測定と測線図の作成を行なう方法をいう。これによりまず人間の手で磁気地図を描き，それを数値化して，コンピュータに記憶させる。コンピュータは伊藤忠電子計算サービスKKのCDC 3600 システムで，フィルタ計算，座標計算を行なったのち，オフラインのCal Comp 763/770 ディジタルプロッタに5万分の1の磁気地図を描かせたものである。

プロッタ・ルーチンは “CONTOUR” とよばれる等高線を描くためのもので，もちろん地形図，気象図，結晶図などにも応用できる。CONTOUR では地図の精度を高めるために，格子点を16点方式でプロットしている。

今回完成した磁気図は能代沖のものであるが，磁気図は海洋開発，資源開発に有用なデータを提供するので，地質調査所では引き続いて佐渡沖の空中磁気探査を開始している。海洋科学の振興が叫ばれている折柄，この方面でもコンピュータの今後の活躍が期待される。

ちなみに，地図といえば，近ごろはやりのドライブマップだのレジャーマップだのを連想するが，いっそのこと，交通量のシミュレーションなどを使って，最も安全かつ最適なドライブマップをコンピュータに画かせてみたら，どうだろう。“最も安全なのは，車を走らせないことである” などという答が出るかどうか？

☆ RTCC システム

月の表面まで 15km に接近したアポロ10号は予定どおり無事帰還したが，宇宙船と地球を結ぶ命綱である「有人宇宙飛行追跡網」と，それが伝送するデータの処理機構について紹介しよう。

宇宙船の追跡網は地上局17個所，船4隻，飛行機8機の陸海空にまたがる布陣であった。追跡網の本拠地はメリーランド州のゴッダード宇宙飛行センターである。ここには，世界最大の IBM システム/360 モデル95が2台と，それに次ぐ規模のモデル91が1台設置されている。モデル95の1台は主として宇宙船の軌道追跡と飛行データ管理のための原始データの処理に使用されている。も

プロッタの多彩な働き **えがかれた磁気地図**

アポロ計画の裏方 **RTCCシステム**

日本へも伸びるか？ **米頭脳会社の触手**

う1台のモデル95と91は宇宙科学研究のために，宇宙の数学モデルを作り出し，銀河・恒星・惑星の起源や進化のシミュレーションに用いられている。

ゴッダード宇宙飛行センターは，それぞれの追跡局がもつ少なくとも2台のメッセージ・スイッチング用コンピュータUNIVAC 1230によりリレーされたメッセージを再チェックし，優先度をつけてヒューストンにある「有人宇宙飛行センター」に集中している。

ヒューストンにある米航空宇宙局のIBMリアルタイム・コンピュータ・コンプレックス（RTCC，本誌4月号，グラビア参照）は，主として次のような計算を行なった。

1）宇宙船を月の周りの放物線軌道に乗せる際，宇宙船が「フリー・リターン軌道」（宇宙船が途中で機械に故障を起こしても，月の裏を周って自動的に地球に帰る軌道）をとるために必要な軌道修正の計算
2）宇宙船が月周回軌道に乗るための軌道修正・逆推進噴射指令計算
3）月着陸船を15kmの高さまで降下させる誘導計算
4）降下・上昇の推進システムの点火指令計算
5）司令船とのランデブーのための方向計算
6）地球へもどるための噴射装置の点火時刻・調節計算
7）大気圏に無事に再突入するための軌道修正計算
8）予定された着水地点へ到達するための計算

このRTCCは，1日に800億回の計算が可能な5台のIBMシステム/360 モデル75で構成されている。

一方，アポロ10号の宇宙船に搭載されたIBM製の計器機構（IU）は，サターン5型ロケット発射前にはロケットの打上げ方向を計算し，また打上げ後は毎秒25回宇宙船に運転命令と姿勢制御命令を与え，地球を周る軌道に導いた。

このIUはまた，第2段ロケット，さらに第3段ロケットに点火命令を下し，エンジンをコントロールしながら，宇宙船を月を周る放物線軌道に導いた。さらに，IUは飛行士が司令船を切り離し，月着陸船とドッキングを行ない，この結合体を第3段ロケットから切り離すのを助け，充分に離れたころ，この第3段ロケットが太陽の周りの軌道に乗るよう推進力をつけるため，残っている燃料を消費するよう命令を下した。

宇宙船が無事軌道に乗ると，月着陸船の下降用エンジンが月の裏側で点火され，宇宙船が表側にきて追跡可能になると，二つのインテルサット（通信衛星）と世界中にはりめぐらされた通信網から送られてきたデータはヒューストンのセンターで処理された。

アポロ計画成功の陰にあるコンピュータの働きについては，かねてから噂にきくところであるが，このような全貌を知らされると，改めてコンピュータの機械としての強烈な個性（？）を痛感せずにはおれない。

☆ 米頭脳会社の触手

ハードウェアの高性能化に伴って，コンピュータ・ユーザーへのサービスで優位に立つためには，ソフトウェアがハードウェアにも増して重視される傾向にあるが，それを裏書きするかのように，米国のソフトウェア会社や情報会社などの欧州進出や日本の企業との提携あるいは合弁会社設立の動きが活発になってきた。

世界最大のソフトウェア会社である米国のコンピュータ・サイエンス社（CSC，従業員3,000人，'68年商5,500万ドル）の首脳が，わが国のビジネス・コンサルタント（資本金1千万円）と合弁のソフトウェア会社の設立折衝のために5月中旬に来日した。

また，同じころ米国有数の調査・研究開発会社であるレアスコ・システムズ・アンド・リサーチ社（従業員1,500人，年商1,000万ドル）は，日本リサーチセンター（資本金7,500万円）に合弁を含む広範な業務提提を申し入れた。

さらに，米国の有力出版社エンサイクロペディア・ブリタニカ社が，東京放送，凸版印刷との間で進めている合弁会社「TBSブリタニカ」（仮称）の設立計画も大詰めを迎えていると噂されている。

さらに5月末には，二つの米有力会社の欧州でのソフトウェア大手会社取得が伝えられた。その一つはダラスのユニヴァーシティ・コンピューティング社（UCC）によるスイスのオートメーション・センターズ・インターナショナル（ACI）の買収で，これによりUCCは欧州ではIBMに次ぐサービス能力をもったことになる。もう一つは，大手銀行のバンカーズ・トラスト社による英国のサイバネティックス・ワールド・トレード（CWT）の株式40％の買収である。

通産省は，現在までに正式に外資導入を申請した企業は一つもないとして楽観的な見方をしているが，しかしひとたび申請が出されたとなると，今後のわが国の情報産業の発展にかかわる重要問題なので，米国情報産業の進出には慎重に取組むものと見られる。

オペレーティング・システムとは

浦　昭二・原田賢一

CPU, I/O, メモリなどのシステム資源（ソリース）の割付けと管理を担うオペレーティング・システムのはたらきを知ろう

はじめに

使用者が計算機を使って問題を処理する場合に，一つのまとまった仕事のことを**ジョブ**（job）という。たとえば，システムに用意されている連立方程式の解を求めるプログラムを呼び出して，それに必要なデータを与えて計算させたり，FORTRAN で書いたプログラムをコンパイルして実行させたり，あるいは給料計算をする場合，それぞれが一つのジョブになる。オペレーティング・システムは計算機でジョブを処理するのに人間の介入をなるべく少なくして，計算機を効率よく動かし，ジョブの処理能力を高めるために作られたプログラムである。

FORTRAN や COBOL のような言語を用いてプログラムを書き，それを実行して正しい結果が得られればよいというような普通の計算機使用者にとっては，オペレーティング・システムは表面に出てこない存在なので，ジョブの内容を知らせる**制御カード**（control card）を通して以外には，関係することがないであろう。しかし，コンパイラやアセンブラなどの言語処理プログラムは，使用者のプログラムを実行するために，すべてこのオペレーティング・システムの制御を受けながら，その助けをかりてジョブを処理しているのである。このようにジョブを直接処理する各種のプログラムはオペレーティング・システムの支配下に置かれ，全体としてソフトウェア・システムが構成されている。

したがって，オペレーティング・システムに関する知識はソフトウェアを勉強しようとする人や計算機を使おうとする人にとっては欠かすことができないものである。以下3回にわたり，オペレーティング・システムについて，発展の動機，その機能，構成と形態，および最近の動向を解説していくことにする。

Shoji Ura, Ken'ichi Harada
慶応義塾大学工学部

1. 計算機使用の能率化
——オペレーティング・システムがつくられるまで

計算機が使われ出したときから，その使い方の能率が問題となり，専門の要員を配置したり，処理の流れが円滑になるような機構にしたり，プログラミングや各種装置の使い方を標準化したりして計算機を有効に働かせようとする努力が払われてきた。これと同時にハードウェアおよびソフトウェアが著しい進歩を示し，これまで人間が行なっていた部分をなるべくプログラムで処理することによって，計算機操作員（オペレーター）の作業を軽減し，以前よりもいっそう計算機を遊ばせないで使用することができるようになった。やがて，このようなプログラムは整備，統合されて，計算機のオペレーションの制御および管理をするシステムへと発展し，オペレーティング・システムという名前でよばれるようになった。以下に電子計算機が一般に使い出されるようになったときから，オペレーティング・システムが作られるに至るまでの移りかわりについて述べる。

1. 初期のころの使い方

1946年アメリカのペンシルバニア大学で世界最初の電子計算機（ENIAC）が作られ，1949年にイギリスのケンブリッジ大学において，今日のようなプログラム内蔵方式をもつ最初の計算機（EDSAC）が作られた。そのころの計算機は現在のものとくらべると処理能力も小さく，

ソフトウェアも発達していなかったため，計算機を使うには次のような作業を一人でしなければならなかった。

1）問題を分析して流れ図を作る

2）流れ図をもとにして機械語あるいは簡単な記号プログラミング言語でコーディングする

3）プログラムやデータを紙テープあるいはカードに穿孔する

4）計算機本体のスイッチやボタンの状態を確認し，使用する入出力装置の準備をする

5）穿孔してきた機械語のプログラムあるいは記号プログラムの翻訳ルーチンを記憶装置に読み込む（**ロードする**）ためのプログラム（ブートストラップ*という）を手操作で操作卓から記憶装置に入れる。ブートストラップを実行することによって，必要なプログラムを記憶装置に読み込む

6）記号プログラミング言語で書いたプログラムの場合には，翻訳プログラムを実行させることによって機械語のプログラムを作る

7）スタートボタンを押して実行を開始させる。もし，使用者の意図した結果が出なかったときには，操作卓のスイッチやボタンを押しながらプログラムの誤りを見つけていく

8）プログラムに誤りの個所を見つけた場合，そこを修正して，5）からの作業を繰り返す

9）計算機を使い終ったら後始末をして次の人に引きわたす

したがって計算機を使うには，機械語の知識，コーディングの技法，各種装置の取扱い方，操作卓の使い方，および異常な事態が発生した場合（停電になったり，ハードウェアが誤動作したりしたとき）の処置の仕方などについて一通り習得していなければならず，使用者はごく少数の人に限られていた。このような計算機の使い方の特徴は，一人の使用者が長時間計算機を占有してしまうことである。その理由としては，**デバッグ**（プログラムの論理的な誤りをみつける作業）が面倒であったことがあげられる。

プログラムは機械語あるいはそれに近いレベルの言語を用いて書いていたために，読みづらく意味のわかりにくいものであった。それだけにコーディングの誤りを犯しやすく，また誤りをみつけるのが困難であった。デバッグにおいては，プログラムの実行に関する情報（記憶装置，レジスタ，あるいはカウンタの値，インディケータの状態）は操作卓からでしか得ることができなかったので，プログラムを書いた本人が操作卓の前に座って機械と1対1でチェックしなければならなかった。その後，プログラムの実行結果を追跡するトレーサや記憶装置の内容を表示したり，変更したりするプログラムが使用されるようになり，デバッグはいくらか容易になった。

2. オペレーティング・システム以前の能率化

やがて，記憶容量の増大，処理装置および入出力装置の高速化，信頼性の向上などハードウェアの進歩にともなって，使いやすいプログラミング・システムが作られるようになり，各種のソフトウェアが整備されてきた。このような発達によって計算機は各分野でさかんに使われ出し，前述のような使い方では能率が悪く，ジョブが処理しきれなくなってきた。

最初に行なわれた計算機使用の能率化は3），4），5），6）の作業で，それぞれ専門要員が置かれるようになった。とくに，専門の計算機操作員を置くことによって，機械操作に不慣れな使用者によって生じるむだな時間は短縮された。しかし，多くの使用者から依頼されるジョブにはさまざまな型があり，ジョブを処理するたびに処理プログラムなどを入れかえたり，入出力装置の準備をしたり，センス・スイッチ**をセットしなおさなければならなかったので，ジョブを処理するための準備に依然として長い時間が費された。このような時間を短縮するためにとられた次の手段は，計算機システムの使い方の標準化である。たとえば，入力テープ，出力テープおよび作業テープの装置番号を約束によって固定させたり，センス・スイッチの使い方を決めたり，操作員に対して出力するメッセージを統一したりして，操作員の作業が機械的になるようにした。

3. オペレーティング・システムの登場

作業方法をいくら改善しても，また操作技術をいくら高めても，人間の動作時間には限りがある。次にとられたのは，操作員の作業をできるだけプログラムで置きかえていく方法である。もし人間でも計算機でも同じことができるとした場合に，人間が行なう手操作の時間と計算機が命令を実行する時間をくらべれば，後者のほうがはるかに短く，また正確であって，ジョブ処理の能率化として当然の方向であることは明らかであろう。たとえば，磁気テープの巻戻しを考えると，人間なら磁気テープ装置のところまで歩いていって，操作ボタンをいくつか押さなければいけないが，計算機なら巻戻し命令を出

---

* boot-strap　カードや紙テープからプログラムを最初に記憶装置に読み込むための数語の命令。

** sense switch　操作員の意思をプログラムに表示するための操作卓のスイッチ。

産報のコンピュータ専門書

最新刊！学校・職場のテキストに最適　絶賛発売中

わかりやすいプログラミング(2)
《COBOL文法の実際》

電子計算機のめざましい発展・普及ぶりは、プログラミングに関する技能の修得が企業に従事するものの必須条件となる日もそう遠くはないのではないかと思わせる。本書は、COBOL文法の基本を順序よく解説するとともに、実務にそのまま役立つような例題を多数選んで、理解と実力を確実に身につけるよう配慮した。練習問題を多数収録・各種学校・職場のテキストとして最適。

ADSコボル研究課編　A5判・228頁　定価720円

わかりやすいプログラミング
岡崎嘉春・角山栄治　¥740

ALGOLの演習
岡崎嘉春・角山栄治・林達也著 ¥620

やさしい電子計算機
渡部弘之著　¥680

電子計算機用語事典
渡部弘之著　¥1,200

《近刊》
わかりやすいプログラミング(3)
わかりやすいプログラミング(4)

産報・東京千代田区神田佐久間町1
振替 東京36786 ☎(253)0111

すだけでよく，わずかの時間ですぐに使える状態にすることができる。プログラムによるオペレーションの能率化は次のような点から進められた。

1）システム・プログラムの読込み：　一般使用者の書いたプログラムを翻訳したり，通訳したりして実行するプログラム，与えられたデータをもとに必要な計算をするプログラム，およびそれらの助けとなるような機能を遂行するプログラムのことを**システム・プログラム**という。これらのプログラムはそれぞれ異なる形式で磁気テープに書かれたり，カードあるいは紙テープに穿孔されたりしているために，記憶装置に読み込むための方法が異なり，操作が面倒であった。そこで，まずシステム・プログラムの形式を統一し，標準的な形式で書かれたシステム・プログラムを読み込むプログラム（**ローダ**）を作ることによって，そのような繁雑さをなくす方法がとられた。このプログラムはひじょうに短く，ジョブを処理しているときにも，こわされないような所に入れられている。操作員は各ジョブの処理に必要なシステム・プログラムを用意しておいて，ローダにプログラムの読込みを指示するだけで，使用可能な記憶場所やインディケータのクリアを行ない，システム・プログラムを読み込んで処理を開始することができる。

2）入出力の能率化とエラー処理：　入出力操作は内部処理にくらべて比較にならないほど遅く，入出力操作をしているときには中央処理装置は遊んでいる状態であった。そこで，入出力操作と内部処理を並行させることによって，いくらかでも能率のよいプログラムの処理ができるようにするため，入出力操作の完了を知らせる**割込み**機能が装備された。入出力に関する割込みには，このほかに読込みや書出しのエラー，入出力装置の動作不能を知らせるものがある。この機能によって正しい入出力ができないときにでも，いちいち操作員の手をかりないでプログラム自身でなるべく処置することができるようになった。

たとえば，磁気テープにキズがあって正しくデータが書けない場合には，キズのない磁気テープにとりかえてプログラムを実行しなおすという必要はなく，その部分を無視して次のキズのない部分にデータを書くようにし，プログラムの実行が中止されることはない。入出力操作はほとんどすべてのプログラムにとって必要な機能であることと，割込みを利用した入出力のプログラムがひじょうに複雑な機構になることから，既成のプログラムをつねに主記憶装置に格納して

おいて，どのプログラムからでも簡単な呼出しによって満足のいく操作ができるようになった。入出力に関係した機能をもち，計算機システムの入出力を集中的に制御し管理するプログラムのことを**入出力制御システム**（Input/Output Control System，略して IOCS）という。

3）システム・プログラムの呼出し：　これまでの方法によっても解決されないものに，ジョブ間で発生する時間がある。すなわち，一つのジョブが終って次のジョブを処理するには，まえに使用したカードデックやシステムテープをとりはずし，次のジョブを処理するための準備時間が必要で，この間計算機は遊んでいる状態にある。このような時間の損失をなくすために，計算センターが保持しているシステム・プログラムを一つのテープやドラムに納めてしまい，ジョブの処理に必要なシステム・プログラムが指定されると，そこから必要なものを主記憶装置に呼び出してきて，コントロールを移すプログラムが作られた。

ジョブの処理が終わると，ふたたびこのプログラムにコントロールが返されて，次のジョブの処理に移るようになっているので，一連のジョブを用意しておけば，つぎつぎに必要なシステム・プログラムが呼び出され，処理が続けられていく。

このように人間の介入をできる限り少なくしてジョブを自動的に，しかも連続的に処理していくプログラムが各計算機システムで作られるようになり，**オペレーティング・システム**とよばれるジョブ処理の制御システムが作られるに至った。最も簡単なオペレーティング・システムにおけるジョブ処理の流れを図1に示す。

さらに，計算機の大型化に伴ってその利用形態が多様化し，使用法が複雑になるにつれてシステムに要求される機能も増大し，オペレーティング・システムはますます大きいものに成長してきた。なお，以上に述べたような目的のために作られたプログラムのことをオペレーティング・システムという以外に，モニタ・システム（monitor system），エグゼクティブ・システム（executive system），あるいはスーパーバイザリ・システム（supervisory system）とよぶことがある。

オペレーティング・システムといっても，一概に定まった機能をもっているものではなく，計算機構成の規模や使用目的によって数百命令しかないものから1万命令を越えるものまでさまざまなものがある。オペレーティング・システムの構成もジョブの処理方式（たとえば，バッチ処理方式，タイムシェアリング方式，多重プログラミング方式，リアルタイム処理方式など）によってそれぞれ異なっている。

2.　オペレーティング・システムのはたらき

上述のように計算機を能率的に動かして，ジョブを順次処理していくために作られたオペレーティング・シス

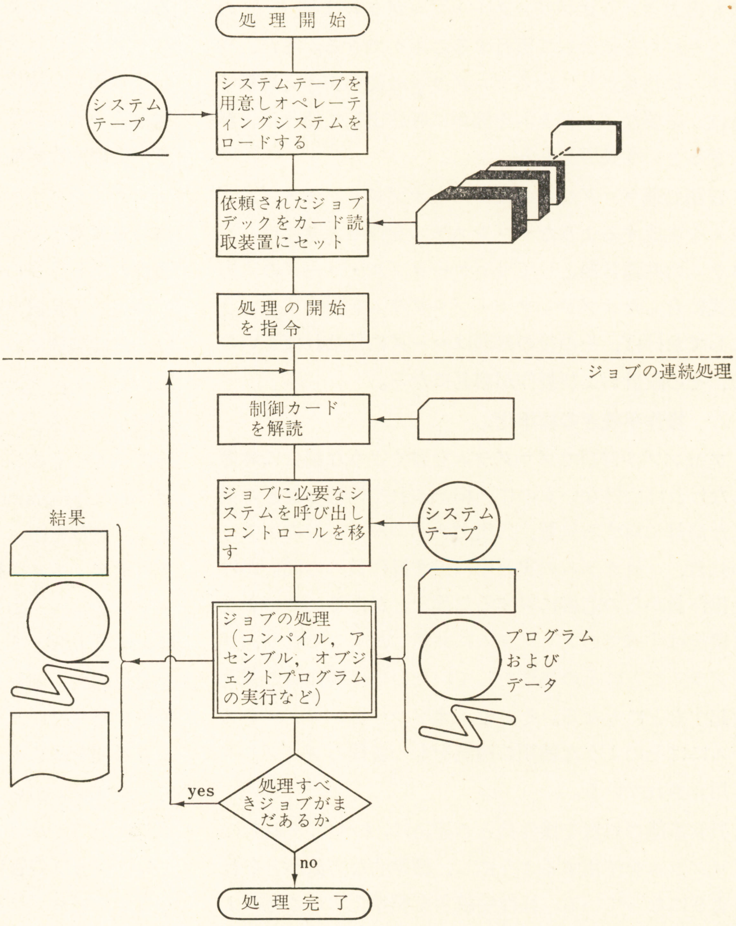


図1　簡単なオペレーティング・システムを用いたジョブ処理の流れ

テムは，その後の新しいハードウェアの開発や計算機の多方面への利用が盛んになるにつれて，しだいに多くの役割をもつようになってきた。以下に，オペレーティング・システムがもつ重要なはたらきをいくつかあげて説明しよう。

1. プログラムの結合

これは独立して作られたプログラムを実行直前に結合して一つのプログラムとしてまとめる機能である。具体的な機能としては，各プログラムが格納される記憶場所の決定，リンケージ（linkage）の編集，共通領域や入出力領域の割当て，システムに登録されているライブラリプログラムの呼出しなどが含まれる。

これらの作業は**リロケータブル・ローダ**，あるいは**リンケージ編集プログラム**とよばれるオペレーティング・システムに組み込まれた専用のプログラムによって処理される。FORTRAN やアセンブラなどの言語処理プログラムが出力するオブジェクト・プログラムの書式を統一することによって，異なる言語で書かれたプログラムでも出力されるオブジェクト・プログラムは同じなので，それらを統合して実行することができる。また，システムに用意されているプログラムの使い方さえ理解すれば，必要なプログラムを簡単に呼び出して使うことができるという長所がある。

言語処理プログラムのなかでもオブジェクト・プログラムを生成するようなサブシステムを作るときには，システムの共通言語としてリロケータブル・ローダの入力形式に合ったオブジェクト・プログラムを出力するようにしておけば，その後の処理はローダに任せればよいので，その設計および製作が容易になる。

2. 操作手続きの標準化

アセンブリ言語でプログラムを書くような場合に考慮しなければいけないものに，桁あふれや桁落ちなどの異常な演算に対する処置，入出力オペレーションのエラーの処置，プログラムの実行中に出す操作員への問いかけと操作員からの応答に対する処理などがある。このような機能はすべてのプログラムに共通したものであり，プログラムごとにその処理をしていたのでは操作員の混乱を起すもとにもなる。そこで，オペレーティング・システムにはそのような種類の機能をもつ各種のプログラムが用意されている。

異常事態の処置や操作員との連絡は，すべて，それらのプログラムを使用するようにし，標準的な方法がとられるようになっている。操作手続きを標準化することによって，操作員は決められた操作法だけ心得ておけばよく，操作はそれほどの判断を要さないで機械的にすればよいので，操作の誤りを少なくすると同時に，操作員が変わるたびにジョブの処理が異なるということもなくなる。

3. システム資源の割付けと有効利用

計算機システムを構成しているハードウェアおよびソフトウェアの要素のことを**システム資源**（system resource）とよび，具体的には中央処理装置，各種入出力装置，主記憶装置，補助記憶装置，各種のプログラムなどが含まれる。オペレーティング・システムはその下で働くコンパイラ，アセンブラなどのシステム・プログラムや使用者のプログラムを実行するとき，あるいは実行中にそれらが必要とするシステム資源の割付けと管理を行なう。すなわち，システム資源の状態（使用可能かどうか，どのジョブで使用されているか）や使用目的に関する情報をたえず保持しておいて，ジョブを処理するときには，そこで必要とする資源を調べて割付けを行なう。

プログラムで入出力装置や補助記憶装置を参照するには，物理的な参照の仕方（チャンネル番号，装置番号あるいはトラック番号など）はとらないで，記号名と必要なパラメータ（記憶容量や使用目的）を与えることによって参照し，実際の対応づけは実行直前に決まるようになっている。こうすることによって，機器が故障したり，物理的な参照の仕方が変わったりしてもオペレーティング・システムが保持しているシステム資源に関する情報を変更するだけで，他に影響を与えないですみ，融通性のある資源の利用ができる。

中央処理装置を有効に使用しようとする技法としては，記憶装置にいくつもの独立したプログラムを入れておいて，入出力の完了待ちのように中央処置装置に遊び時間が生じると，プログラムの実行を適宜切りかえ，同時にいくつものジョブを処理していく**多重プログラミング**（mult-programming）とよばれるジョブの処理方式がある。処理を待っている一連のジョブは，ふつう磁気ドラムやディスクに格納しておかれる。オペレーティング・システムは，山積みされたジョブにおいてそれぞれ必要とする資源とその時点でシステムで使用できる資源とを見比べて，満足させることができるジョブを選択して資源を割り付け，そのプログラムを主記憶装置に呼び込んで処理をはじめる。このような処理方式の採用によって，システムがもっている資源をいっそう有効に利用することができる。

4. ジョブの管理

自分でプログラムを書き，計算機で処理した経験をもっている人にはおわかりのことと思うが，プログラムは

一度書いただけで正しい結果が得られるということはほとんどない。いくら経っても何の応答もなかったり，無意味な出力が限りなく出てきたり，重要なデータをこわしてしまったり，突然計算機の動作が止まってしまったりということがしばしばある。このようなときには，プログラムを修正して再びシステムをロードし，実行しなければならない。

オペレーティング・システムはこのような事態の発生や，割り付けられた資源の矛盾した使い方をつねに監視していて，そういったことからシステムの破壊を防ぎ，計算機のむだな動きをなくす役割ももっている。さらに，多重プログラミング・システムのように高度なオペレーティング・システムになると，プログラムの誤りによって，システムや他の使用者が所有しているプログラムやデータがこわされることは絶対に避けなければならないので，この監視はハードウェアの助けをかりていっそう厳密なものとなってくる。

5. 処理の記録

ジョブを処理していくときに，ジョブの依頼者名，受付けコード，処理時間，ラインプリンタやカード穿孔装置などへの出力量，ジョブ打切りのときにはその理由などを記録しておくことは，計算機の使用料金を計算したり，計算機を管理したりするのに重要である。

簡単なオペレーティング・システムであれば，これらの作業は操作員がしなければならないこともあるが，ふつうのオペレーティング・システムにおいてはジョブの処理を開始するときに与えられる制御カードから必要な情報をとり出しておいて，処理が終わると操作員用の**コンソール・タイプライタ**あるいは処理記録をとるための専用の装置(紙テープやディスク)に出力するようになっている。ジョブを1件1件処理していくようなシステムにおいては操作員がこのような記録をとることもできるけれども，システム内に同時にいくつものジョブをかかえて処理していくようなときには正確な記録をとることができず，ぜひともオペレーティング・システムにこのような記録の機能を備えておくことが必要とされる。

6. 処理の中断・再開およびバックアップ

数時間もかかるような長いジョブを処理している最中に緊急のジョブが発生して，いまやっているジョブを中断して新しいジョブを処理しなければならなくなったり，ハードウェアの故障によって処理が中断されてしまう場合がある。このようなときのために，オペレーティング・システムは，ある時点におけるシステムの状態を外部記憶装置に記録しておいて，後刻，そのときの状態を再現し，実行を再開させることができる機能を備えている。

ジョブ処理の中断は操作員の判断によってコンソール・タイプライタから指示されたり，異常事態の発生を知らせるハードウェアの信号が出されたりすることによって行なわれる。1回の処理に何時間もかかり，計算機が止まるたびに最初からやりなおしていたのではいつ終わるかわからないようなジョブの場合には，オペレーティング・システムに対して実行中のプログラムからも状態の保存を要求することができる。このような情報は通常磁気テープあるいは磁気ドラムにはき出されるようになっている。

銀行の窓口業務，座席予約，商取引の自動処理のように重要な情報を扱う場合には万一のときに備えて，ある時間間隔でシステムの状態の保存を行ない，何かあったときにはその時点になるべく近い時点の状態が再現できるような**バックアップ体制**がとられている。このような所で扱う情報はひじょうに大量であり，磁気ディスクなどの大容量記憶装置の内容を磁気テープに書き出したとすると1回で何本ものテープになってしまって，その保管もたいへんであるが，バックアップ用の記録をとるのに多くの時間を費してしまうことになる。そこで前回の書出しのときと比較して変化のあった部分だけをその識別情報とともに書くようにして，システムからはき出す情報の量を減らし，そのような手続きに時間をかけないようにしているところもある。

以上で，オペレーティング・システムが作られるようになるまでの背景とオペレーティング・システムはどんなことを行なうプログラムであるかについて簡単に述べてきたが，次回は簡単なオペレーティング・システムをとりあげて，その構成とジョブの処理について解説していく予定である。

科学と随想
蟻塔 ■7月号
通巻 165
vol.15
No. 7
■エジプトのイメッジ今昔　佐藤孝己
■マスプロ教育の問題点　東　健一
■鉄砲と歌舞伎　千藤三千造
■調和と純粋さ　和田淳蔵
■Kühn 教授の思い出　内田　亨
■続・ケンブリッジあちこち(6)　武田和子
購読ご案内
■購読ご希望の方は、住所、氏名、年令ご専門をご記入のうえ、購読料2年分240円を切手にてお払い込み下さい。
共立出版株式会社

質問 アセンブラ言語の参考書は？

小生は，現在九州電子計算機学校に通学して勉強中の者ですが，アセンブラ言語についてとくに詳しく知りたいと思います。どのような参考書があるか教えてください。 (福岡市Y. N. 病院勤務)

答え 各機種のマニュアルが手掛り

各電子計算機に共通なアセンブラ言語は皆無であるといえますが，わずかに，かつて日本電子工業振興協会のSIP委員会で開発したSIPというアセンブラ言語があります(SIPというのは，Symbolic Input Programmingの略です)。このSIPは国産電子計算機を対象として，ある程度共通に使用できることを目的として作られたものですが，現在はあまり使用されていないようです。

また，電子計算機ごとに命令とその機能がそれぞれ異なるので機械語と1対1に対応するアセンブラ言語では，とくに機種によってひじょうに異なると考えなければなりません。したがって実用的なアセンブラ言語に関するものであれば，各電子計算機メーカーが作っている，各機種のアセンブラ言語の使用説明書を参照すべきでしょう。

一般に市販されている書物でアセンブラ言語についてだけ書かれているものはなく，プログラミングの説明と，言語を自分で設定して，その使い方を説明しているものがほとんどです。

下記は一般に市販されている書物の一部ですが，このうち，1)，2)，3)はSIPを使って説明されています。

1) 「電子計算機のプログラミング」<オートメーションシリーズ> 6 (共立出版)
2) 清水辰次郎著「実用数学」朝倉数学講座，第20巻(朝倉書店)
3) 森口繁一著「電子計算機」(筑摩書房)
4) P. M. シャーマン著・関根智明訳「電子計算機—プログラミングとコーディング」(竹内書店)
5) 一松　信著「電子計算機のプログラミング」(日本評論社)
6) 藤川洋一郎・赤　摂也著「電子計算機入門」新数学シリーズ，25 (培風館)

ドクタービット Dr. bit

質問 資格認定制度についてのお尋ね

電子計算機学校の広告欄に「EDPプログラマー2級の認定証」というようなことが書いてありますが，この認定証はどういう規準のものでしょうか。資格認定制度はまだ確立されていないようですが。 (東京都T. K. 学生)

答え まだ民間試験の段階です

全般的な資格試験について，現在実施しているものを大別すると，法律の規定に基づいて行なっている「国家試験」と，民間団体などが独自の基準を定めて実施している「民間試験」とがあります。民間試験の中でも，商工会議所・商工会などが行なう珠算・簿記検定など準国家試験的なものを除けば実施団体乱立の嫌いがあり，資格認定基準が必ずしも統一されておりません。

情報処理技術者においても同様で，新しく生まれた職業でもあり，職種の分類・技術水準などを明確にすることがむずかしく，各種学校，通信教育および特定の講習会などでは，いずれもその受講者のみに，その主催する団体独自の基準により認定証または修了証などを発行しているのが実状です。

質問内容にあります「EDPプログラマー2級の認定証」はこの部類にはいるものです。また，一般からの公募の形で，だれでも受験でき，合格者のみに認定を行なっている団体もありますが，残念ながら世間一般にその実力を認められていないようです。

以上のような問題からも，今後ますます急増すると思われる情報処理専門職員の資質の向上とその社会的地位の確立をはかる必要性が各界より要求されており，政府通産省においても，情報処理部門の認定試験制度の実施を目的として，検討を行なっているのが現状です。

なお，一般からの公募により認定制度を実施しているものとして，下記の(社)日本事務能率協会データプロセッシング資格試験委員会が認定する「データ・プロセッサー」および「初級データ・プロセッサー」の二つが中でもよく知られています。

(社)日本事務能率協会・データ・プロセッシング資格試験委員会
東京都渋谷区千駄谷4—25—2 (郵便番号151)
TEL 東京 (403) 1331

科学用技術計算から計測データ処理まで
守備範囲は極めて広汎です

ＩＣの発達と量産の技術の進歩が生んだ高性能コンピュータHITAC 10の魅力は、個人で自由に使えること——持ち運びが簡単で、100Ｖの電源でもＯＫという手軽さです。しかも、小形ながら中形機なみの高性能を秘めています。活用範囲は、科学技術計算はもちろん、計測データ処理やハイブリッド計算など、あらゆる用途に最適です。サイクルタイム1.4μs、加減算2.8μsで記憶容量は４～８Ｋ語という性能も、大きな魅力。ソフトウエアとしては、フォートランやカリキュレータが、４Ｋ語の基本構成で使えます。

HITAC 10 新発売

価格495万円（リースのお取扱いもいたします）

資料および詳細は、電子機器第一部コンピュータ課または、もよりの営業所へお問合わせください。

日製産業株式會社

本　社　東京都港区西新橋2－15－12(日立愛宕別館) 電話 東京(03)503－2311(大代)〒105
営業所　大阪(06)363－3331・名古屋(052)581－6211・福岡(092)75－2936・広島(0822)21－4511・仙台(0222)25－5561・札幌(0122)22－7241・富山(0764)41－3386

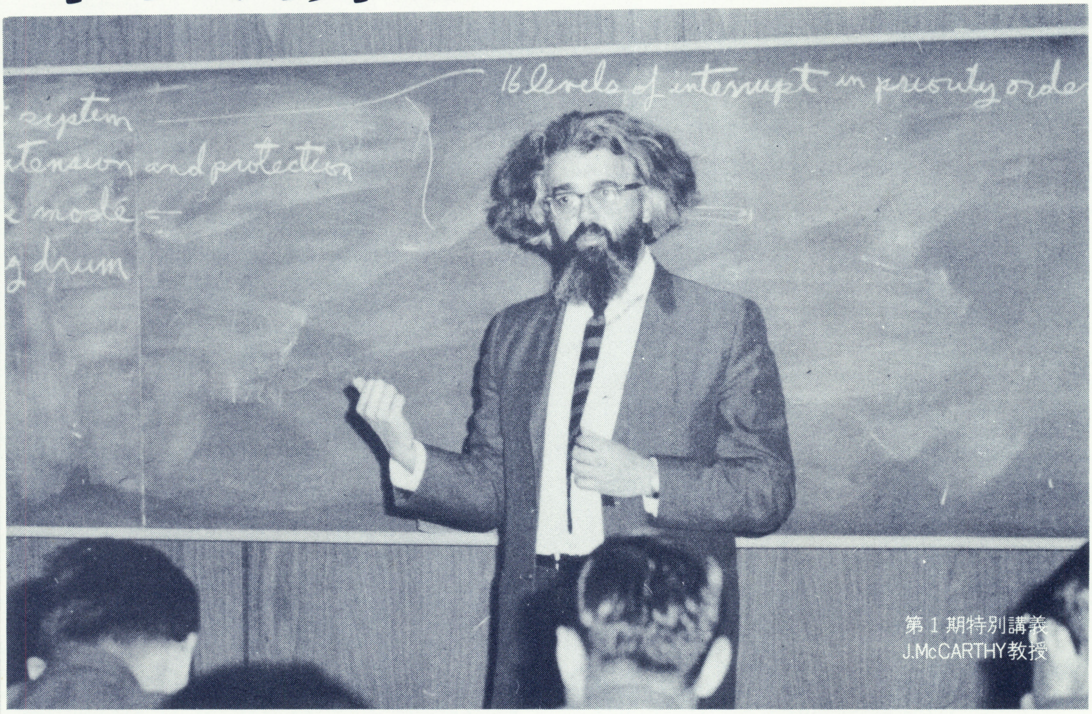
株式會社　日立製作所

上級情報処理技術者のための

JIPDEC
中央研修所

第２期研修生募集

受付期間　７月１日(火)～７月19日(土)



第１期特別講義
J.McCARTHY教授

研修コース	コンピュータ・スペシャリストコース（下記コース別受講も可能） ●プログラミングコース　●経営科学コース ●システム設計コース　●特別講義
募集人員	30名
期　　間	昭和44年８月４日（月）～12月12日（金）
受　講　料	一般15万円，賛助会員12万円（全コース受講の場合）
応募資格	大学卒程度の学力を有し、企業経験３年以上 プログラミングの実務経験１年以上
場　　所	東京都港区芝公園機械振興会館内　中央研修所

昨年10月開講された日本情報処理開発センター中央研修所の第１期研修生40名は本年３月所定の研修を終りそれぞれの会社で活躍中であります。
中央研修所の第２期研修はよそおいを改めてきたる８月４日開講いたします。
この研修所は急速に進む情報処理利用技術の基本を体得し、情報処理システムの開発、設計、運営における中核的、上級技術者の養成を目的としたものであります。

（詳細については募集案内をお送りしますから下記へ御連絡下さい）

財団法人　**日本情報処理開発センター**　技術部教育課

東京都港区芝公園21号地 機械振興会館内　〒105
電　話（434）８２１１（大代表）　内線 475

連載　線形計画法⑤

感度分析とパラメトリックLP

藤川洋一郎

基礎データの値をパラメタ（変動値）と考えて，線形計画問題を再考してみよう

前回および前々回の単体法の説明では，投入産出係数や総投入量や利益係数の値はある確定した値，つまり定数として扱ってきた。しかし実際にはこれらの値は統計資料に基づいて推定される値であり，むしろある幅の誤差変動を伴う統計量と考えるべきである。また場合によっては新設計画中の活動に対する設計値であることもある。

このような場合，基礎データ値の確定した一つの模型についてだけ最適計画や帰属価格を求めるのでは不充分であり，最適計画や帰属価格が基礎データ値の変動にどのように影響されるかを分析することが重要になるし，また，これらのデータ値は定数ではなくむしろパラメタとして扱い，このパラメタの変動に伴う最適計画や帰属価格の変化を追跡するような手段も必要になる。

1.　利益係数の変化

まず最初に利益係数の変化が最適計画にどのような効果をおよぼすかを単体法の説明の際に扱った例について分析してみよう。与えられた問題の解はまとめると次のようであった。

生産活動（＊印は基底活動）	生 産 量（トン）	限界価値（万円）	利益係数（万円/トン）
A	0	$-\frac{3}{8}$	3
B＊	20	0	2
C＊	20	0	4
D	0	$-\frac{3}{8}$	3

活動要素	総投入量	帰属価格
設備容量	400時間	$\frac{1}{8}$（万円/時間）
原　　料	280トン	$\frac{1}{4}$（万円/トン）

総利益＝120万円

この最適計画において基底活動にはいっていない活動AとBに注目すると，その限界価値の示すところによればいずれも生産量1トン増当たり$\frac{3}{8}$万円の利益減をもたらすことがわかるが，これは利益係数が現在の値3万円/トンよりある程度増加した場合は事情は変わってくるのであって，むしろ現在の基底活動BやCに比べて活動AまたはDのほうが有利になって代わりに基底活動に取り入れたほうがもっと総利益が増すことが考えられる。ではいったいどれだけ活動AまたはDの利益係数が増えるとこのような事情の変化が起こりうるか。この問題は限界価値と利益係数との関係に注目すれば簡単に解決される。活動AとDについていえば限界価値 $\bar{p}_1$，$\bar{p}_4$ と利益係数 p_1，p_4 の間には次のような関係が成立することを想起しよう。

$$\bar{p}_1 = p_1 - u_B a_1 \qquad (1_1)$$

$$\bar{p}_4 = p_4 - u_B a_4 \qquad (1_2)$$

ここに u_B は現在の基底計画に対応する帰属価格ベクトルであり，a_1，a_4 はそれぞれ活動AとDの投入産出係数ベクトルであって，いずれも p_1 や p_4 には無関係な定数ベクトルである。したがって p_1，p_4 の変化はそれぞれ独立に $\bar{p}_1$ と $\bar{p}_4$ だけに変化をおよぼすことがわかる。そこで p_1 が θ だけ増加すれば $\bar{p}_1$ も θ だけ増加することになる。$p_1=3$（万円/トン）に対して $\bar{p}_1=-\frac{3}{8}$（万円/トン）であるから $p_1=3+\theta$ に対して $\bar{p}_1=-\frac{3}{8}+\theta$

Yōichirō Fujikawa
立教大学理学部

となる。そして

$$\bar{p}_1=-\frac{3}{8}+\theta\leqq 0$$

をみたす限り，すなわち

$$-\infty<\theta\leqq\frac{3}{8}$$

をみたす限り，$\theta=0$ のときの最適計画が依然として最適である。

一般にある最適計画において，その非基底活動 A_i の限界価値を $\bar{p}_i$ とすれば，その最適計画がその活動の利益係数 p_i の増分 θ に対して依然として最適計画となるためには θ は次の条件をみたさねばならない。

$$-\infty<\theta\leqq-\bar{p}_i \qquad (2)$$

次に基底活動の利益係数が変化する場合の影響を考えてみよう。基底活動の利益係数ベクトル $\boldsymbol{p}_B$ と帰属価格ベクトル $\boldsymbol{u}_B$ との間には

$$\boldsymbol{u}_B=\boldsymbol{p}_B\boldsymbol{B}^{-1} \qquad (3)$$

という関係が成立することを想起しよう。上例の最適計画において $\boldsymbol{p}_B=[p_3,\ p_2]=[4,\ 2]$ であるが，いま p_3 が θ だけ増加した場合の $\boldsymbol{p}_B$ を $\boldsymbol{p}'_B$ と表わせば

$$\boldsymbol{p}'_B=[4+\theta,\ 2]=\boldsymbol{p}_B+\theta[1,\ 0]$$

となり，これに対応する $\boldsymbol{u}_B$ を $\boldsymbol{u}'_B$ とすれば

$$\boldsymbol{u}_B'=\boldsymbol{p}'_B\boldsymbol{B}^{-1}=\boldsymbol{p}_B\boldsymbol{B}^{-1}+\theta[1,0]\boldsymbol{B}^{-1}$$

$$=\boldsymbol{u}_B+\theta\begin{bmatrix}\frac{3}{32} & -\frac{1}{16}\end{bmatrix} \qquad (4)$$

となる。さらに，$\bar{p}_1$，$\bar{p}_4$ の新しい値を $\bar{p}'_1$，$\bar{p}'_4$ とすれば

$$\bar{p}_1'=p_1-\boldsymbol{u}'_B\boldsymbol{a}_1=p_1-\boldsymbol{u}_B\boldsymbol{a}_1-\theta\begin{bmatrix}\frac{3}{32} & -\frac{1}{16}\end{bmatrix}\boldsymbol{a}_1$$

$$=\bar{p}_1-\theta\begin{bmatrix}\frac{3}{32} & -\frac{1}{16}\end{bmatrix}\boldsymbol{a}_1$$

$$=-\frac{3}{8}-\theta\begin{bmatrix}\frac{3}{32} & -\frac{1}{16}\end{bmatrix}\begin{bmatrix}3\\12\end{bmatrix}$$

$$=-\frac{3}{8}+\theta\frac{15}{32} \qquad (5)$$

$$\bar{p}_4'=p_4-\boldsymbol{u}_B'\boldsymbol{a}_4=p_4-\boldsymbol{u}_B\boldsymbol{a}_4-\theta\begin{bmatrix}\frac{3}{32} & -\frac{1}{16}\end{bmatrix}\boldsymbol{a}_4$$

$$=\bar{p}_4-\theta\begin{bmatrix}\frac{3}{32} & -\frac{1}{16}\end{bmatrix}\begin{bmatrix}9\\9\end{bmatrix}$$

$$=-\frac{3}{8}-\theta\frac{9}{32} \qquad (6)$$

したがって

$$\bar{p}_1'=-\frac{3}{8}+\theta\frac{15}{32}\leqq 0 \qquad (7_1)$$

$$\bar{p}_4'=-\frac{3}{8}-\theta\frac{9}{32}\leqq 0 \qquad (7_2)$$

が成立するような θ である限り最適計画は維持される。

(7_1) から

$$-\infty<\theta\leqq\frac{3}{8}\Big/\frac{15}{32}=\frac{4}{5}$$

(7_2) から

$$-\frac{3}{8}\Big/\frac{9}{32}=-\frac{4}{3}\leqq\theta<\infty$$

となるのでけっきょく，最適計画が維持される θ の範囲は

$$-\frac{4}{3}\leqq\theta\leqq\frac{4}{5} \qquad (8)$$

ということになる。利益係数の範囲に直せば

$$\frac{8}{3}\leqq p_3'\leqq\frac{24}{5} \qquad (9)$$

となる。もし利益係数 p_3' が $\frac{24}{5}$（万円/トン）を越えるときは (7_1) から $\bar{p}_1'>0$ となり，活動Aを基底活動に取り入れることによって基底計画を改良する余地が生ずる。また $\frac{8}{3}$（万円/トン）より小さくなるときは (7_2) より $\bar{p}_4'>0$ となるので，こんどは活動Dを基底活動に取り入れることによって計画を改良する余地が生じてくる。

利益係数の変化に伴う目的関数の値の変化はどうなるかというと，最適計画において目的関数は基底変数（基底活動水準）だけの一次関数であって，その係数が利益係数であるから，非基底活動の利益係数の変化はその最適計画が維持される限り目的関数の値には影響を与えない。基底活動の利益係数の増分を θ とし，最適計画におけるその活動水準を x とすれば，目的関数値の増分 Δz は

$$\Delta z=x\cdot\theta$$

によって与えられることは明らかであろう。たとえば活動Cの利益係数の増分 θ に伴う目的関数の増分 Δz は

$$\Delta z=20\theta$$

となり，トン当たり1万円の利益係数の増分に対して20万円の総利益増分を伴うことになる。$\theta=\frac{4}{5}$ とおけば $\Delta z=20\times\frac{4}{5}=16$ となる。すなわち製品BとCのみそれぞれ20トン生産して120万円の総利益を得る最適計画は製品Cの利益係数 p_3 が4（万円/トン）から4.8（万円/トン）に増加するまでは依然として最適計画であって，$p_3=4.8$ に達したときその総利益は $120+16=136$（万円）になっている。

さらに p_3 が増加すると生産活動Aを採用して基底計画を改良しない限り，もはや活動BとCの組合せは最適計画ではなくなる。そこで活動Aを基底活動に取り入れることにする。その代わり活動BとCのうち一方を選んで基底活動から追い出さねばならない。この選択は単体法によって行なえる。取り入れるベクトルは $\boldsymbol{a}_1$ であるから

$$t_1 = B^{-1}a_1 = \begin{pmatrix} \frac{3}{32} & -\frac{1}{16} \\ -\frac{1}{8} & \frac{1}{4} \end{pmatrix} \begin{pmatrix} 3 \\ 12 \end{pmatrix} = \begin{pmatrix} -\frac{15}{32} \\ \frac{21}{8} \end{pmatrix}$$

a_2 を追い出し，新しい基底行列 $B=[a_3,\ a_1]$ について

$$B^{-1} = \begin{pmatrix} \frac{1}{14} & -\frac{1}{56} \\ -\frac{1}{21} & \frac{2}{21} \end{pmatrix} \qquad X_B = B^{-1}d = \begin{pmatrix} \frac{165}{7} \\ \frac{160}{21} \end{pmatrix}$$

$$u_B = p_B B^{-1} = \begin{bmatrix} \frac{24}{5} & 3 \end{bmatrix} \begin{pmatrix} \frac{1}{14} & -\frac{1}{56} \\ -\frac{1}{21} & \frac{2}{21} \end{pmatrix} = \begin{bmatrix} \frac{1}{5} & \frac{1}{5} \end{bmatrix}$$

$$\bar{p}_2 = p_2 - u_B a_2 = 2 - \begin{bmatrix} \frac{1}{5} & \frac{1}{5} \end{bmatrix} \begin{bmatrix} 4 \\ 6 \end{bmatrix} = 0$$

$$\bar{p}_4 = p_4 - u_B a_4 = 3 - \begin{bmatrix} \frac{1}{5} & \frac{1}{5} \end{bmatrix} \begin{bmatrix} 9 \\ 9 \end{bmatrix} = -\frac{3}{5} < 0$$

$$z = u_B d = \begin{bmatrix} \frac{1}{5} & \frac{1}{5} \end{bmatrix} \begin{bmatrix} 400 \\ 280 \end{bmatrix} = 136$$

以上で新しい利益係数ベクトル $p = \begin{bmatrix} 3 & 2 & \frac{24}{5} & 4 \end{bmatrix}$ に対して，活動C，Aの組合せによる計画が最適計画になることが確かめられた。そこでさらに p_3 を増加させてどこまでこの最適計画が維持されるかを調べてみよう。

$$p_B = \begin{bmatrix} \frac{24}{5}, & 3 \end{bmatrix}, \quad p_B' = \begin{bmatrix} \frac{24}{5}+\theta, & 3 \end{bmatrix}$$

とおく。

$$u_B' = p_B' B^{-1} = u_B + \theta[1,\ 0]B^{-1} = u_B + \theta\begin{bmatrix} \frac{1}{14}, & -\frac{1}{56} \end{bmatrix}$$

$$\bar{p}_2' = p_2 - u_B' a_2 = \bar{p}_2 - \theta\begin{bmatrix} \frac{1}{14}, & -\frac{1}{56} \end{bmatrix} \begin{bmatrix} 4 \\ 6 \end{bmatrix} = -\frac{5}{28}\theta$$

$$\bar{p}_4' = p_4 - u_B' a_4 = \bar{p}_4 - \theta\begin{bmatrix} \frac{1}{14}, & -\frac{1}{56} \end{bmatrix} \begin{bmatrix} 9 \\ 9 \end{bmatrix} = -\frac{3}{5} - \frac{27}{56}\theta$$

これからわかるように $\theta>0$ である限り $\bar{p}_2'<0$, $\bar{p}_4'<0$ となり， p_3 はどんなに増加しても最適計画は変わらない。ただし目的関数の値 z は

$$z = 136 + \frac{165}{7}\theta$$

となり，θ の増加とともに増加する。

また p_3 を4（万円/トン）から減少する場合についても同様な考察が行なえる。この場合は p_3 が $\frac{8}{3}$ にいたるまでは活動C，Bの組合せが最適であるが，p_3 が $\frac{8}{3}$ を越えて減少するやいなや活動Dの限界価値の符号は正に変わる。そこで活動Dを基底に取り入れるように改良する。その結果は活動Bが基底から追い出されることになり，活動C，Dの組合せが，$p_3=\frac{8}{3}$ に対する最適計画になることが確かめられる。このときの基底逆行列は

$$B^{-1} = \begin{pmatrix} \frac{1}{8} & -\frac{1}{8} \\ -\frac{1}{9} & \frac{2}{9} \end{pmatrix}$$

$$X_B = \begin{pmatrix} 15 \\ \frac{160}{9} \end{pmatrix}$$

$$u_B = p_B B^{-1} = \begin{bmatrix} \frac{8}{3}, & 3 \end{bmatrix} \begin{pmatrix} \frac{1}{8} & -\frac{1}{8} \\ -\frac{1}{9} & \frac{2}{9} \end{pmatrix} = \begin{bmatrix} 0, & \frac{1}{3} \end{bmatrix}$$

$$z = u_B d = \frac{280}{3}$$

$$\bar{p}_1 = p_1 - u_B a_1 = 3 - \begin{bmatrix} 0, & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 3 \\ 12 \end{bmatrix} = -1 < 0$$

$$\bar{p}_2 = p_2 - u_B a_2 = 2 - \begin{bmatrix} 0, & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 4 \\ 6 \end{bmatrix} = 0$$

ここでさらに p_3 を $\theta(>0)$ だけ減少することを考えてみると

$$\bar{p}_1' = \bar{p}_1 + \theta\begin{bmatrix} \frac{1}{8}, & -\frac{1}{8} \end{bmatrix} \begin{bmatrix} 3 \\ 12 \end{bmatrix} = -1 - \frac{9}{8}\theta < 0$$

$$\bar{p}_2' = \bar{p}_2 + \theta\begin{bmatrix} \frac{1}{8}, & -\frac{1}{8} \end{bmatrix} \begin{bmatrix} 4 \\ 6 \end{bmatrix} = -\frac{1}{4}\theta < 0$$

となり， p_3 をこれ以上どんなに減少させても最適計画の組合せは変わらない。このとき目的関数は

$$z = \frac{280}{3} - 15\cdot\theta$$

以上の分析結果を図にまとめると図1のようになる。

2. 総投入量の変化

次に総投入量の変化が最適計画に与える影響を分析してみよう。ふたたび前述の例で活動C，Bを基底活動と

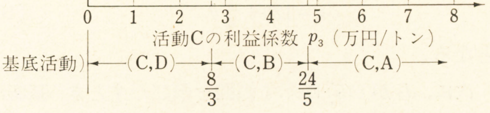


図 1

する最適計画を出発点として考えることにする。総投入量ベクトル $\boldsymbol{d}$ は帰属価格ベクトル $\boldsymbol{u}_B$, したがって限界価値には関係がないが

$$\boldsymbol{x}_B=\boldsymbol{B}^{-1}\boldsymbol{d} \quad (10)$$

$$z=\boldsymbol{u}_B\boldsymbol{d} \quad (11)$$

の関係によって最適計画の基底活動水準ベクトル $\boldsymbol{x}_B$ と目的関数 z にのみ影響をもつことになる。設備容量と原料の総投入量の帰属価格はそれぞれ $\frac{1}{8}$(万円/時間), $\frac{1}{4}$(万円/トン)であるので，設備容量の拡張1時間当たり $\frac{1}{8}$万円の総利益増と原料総入量の増加1トン当たり$\frac{1}{4}$万円の総利益増がもたらされることがすぐわかる。いま原料投入量の増加だけを考えるとして，その増分を θ トンとし，それに伴う $\boldsymbol{d}$, $\boldsymbol{x}_B$, z をそれぞれ $\boldsymbol{d}'$, $\boldsymbol{x}_B'$, z' で記すことにすれば

$$\boldsymbol{x}_B'=\boldsymbol{B}^{-1}\boldsymbol{d}'=\boldsymbol{B}^{-1}\left(\boldsymbol{d}+\begin{bmatrix}0\\1\end{bmatrix}\theta\right)=\boldsymbol{x}_B+\boldsymbol{B}^{-1}\begin{bmatrix}0\\1\end{bmatrix}\theta$$

$$=\boldsymbol{x}_B+\begin{pmatrix}-\frac{1}{16}\\ \frac{1}{4}\end{pmatrix}\theta=\begin{pmatrix}20-\frac{1}{16}\theta\\ 20+\frac{1}{4}\theta\end{pmatrix} \quad (12)$$

となる。限界価値 $\bar{p}_1$, $\bar{p}_4$ は不変であるから最適計画が維持されるためには $\boldsymbol{x}_B'$ が非負条件をみたしさえすればよい。

そのためには(12)から $\theta>0$ について

$$20-\frac{1}{16}\theta\geqq 0 \quad \text{あるいは} \quad \theta\leqq 320 \quad (13)$$

をみたすことが必要である。また θ が(13)の条件を破って増加するやいなや活動Cの水準は負となり実行不可能な計画になってしまう。そこで活動Cを基底活動より追い出して計画を修正しない限り，θ はこれ以上増やせない。また $\theta<0$ の場合については

$$20+\frac{1}{4}\theta\geqq 0 \quad \text{あるいは} \quad -80\leqq\theta \quad (14)$$

をみたす限り，実行可能であり，この条件を越えて θ が減少すると活動Bの水準が負となり実行不可能な計画になってしまう。これ以上 θ を減らすことは活動Bを基底活動より追い出して計画を修正しない限り望めない。けっきょく，原料総投入量 d_2 が

$$200\leqq d_2\leqq 600 \quad (15)$$

の範囲にある限り，活動C，Bによる組合せが最適計画となり，(11)から目的関数は

$$z=120+\frac{1}{4}\theta \quad (16)$$

となる。

次に，d_2 が(15)に示す範囲を飛びだすとき，基底活動の入れかえを行なって計画を修正しなければならない。たとえば $d>600$ となるときは活動Cを基底から追い出して，かわり活動AまたDのいずれかを取り入れなければならない。このとき新しい基底計画によって定まる限界価値はどの活動についても正であってはならない。そこでこのような要求をみたすように取り入れる活動を選ばねばならない。

新しく基底に取り入れる活動の投入産出係数ベクトルを $\boldsymbol{a}_s$ とする。上の例では $s=1$ または4である。もとの基底行列を $\boldsymbol{B}$, 新しい基底行列を $\overline{\boldsymbol{B}}$ とすれば

$$\boldsymbol{B}=[\boldsymbol{a}_3,\ \boldsymbol{a}_2],\ \overline{\boldsymbol{B}}=[\boldsymbol{a}_s,\ \boldsymbol{a}_2] \quad (17_1)$$

$$\boldsymbol{u}_B\boldsymbol{B}=\boldsymbol{p}_B=[p_3,\ p_2],\ \boldsymbol{u}_{\overline{B}}\overline{\boldsymbol{B}}=\boldsymbol{p}_{\overline{B}}=[p_s,\ p_2] \quad (17_2)$$

でなければならない。ところで

$$\overline{\boldsymbol{B}}=\boldsymbol{B}+[\boldsymbol{a}_s-\boldsymbol{a}_3,\ \boldsymbol{0}] \quad (18)$$

ここに $\boldsymbol{0}$ は零ベクトル。(17₂)と(18)から

$$\boldsymbol{u}_{\overline{B}}-\boldsymbol{u}_B\boldsymbol{B}=\boldsymbol{u}_{\overline{B}}\boldsymbol{B}-\boldsymbol{u}_B\boldsymbol{B}+[\boldsymbol{u}_{\overline{B}}(\boldsymbol{a}_s-\boldsymbol{a}_3),\ 0]=\boldsymbol{p}_{\overline{B}}-\boldsymbol{p}_B \quad (19)$$

一方

$$\boldsymbol{p}_{\overline{B}}-\boldsymbol{p}_B=[p_s,\ p_2]-[p_3,\ p_2]=[p_s-p_3,\ 0]$$

であるから(19)と(17₂)から

$$\boldsymbol{u}_{\overline{B}}\boldsymbol{B}=\boldsymbol{u}_B\boldsymbol{B}+(p_3-\boldsymbol{u}_{\overline{B}}\boldsymbol{a}_3)[1,\ 0]$$

あるいは

$$\boldsymbol{u}_{\overline{B}}=\boldsymbol{u}_B+\bar{p}_3'[1,\ 0]\boldsymbol{B}^{-1} \quad (20)$$

ここに $\quad \bar{p}_3'=p_3-\boldsymbol{u}_{\overline{B}}\boldsymbol{a}_3 \quad (21)$

は新しい基底計画における活動Cの限界価値であって，要求に従ってこれは正であってはならない。また活動C以外の活動の新しい限界価値は(20)を用いれば次のようになる。

$$\bar{p}_j'=p_j-\boldsymbol{u}_{\overline{B}}\boldsymbol{a}_j=p_j-\boldsymbol{u}_B\boldsymbol{a}_j-\bar{p}_3'[1,\ 0]\boldsymbol{B}^{-1}\boldsymbol{a}_j$$
$$=\bar{p}_j-\bar{p}_3't_{1j}\quad(j\neq 3) \quad (22)$$

ここに

$$\boldsymbol{t}_j=\begin{bmatrix}t_{1j}\\t_{2j}\end{bmatrix}=\boldsymbol{B}^{-1}\boldsymbol{a}_j$$

また $\bar{p}_j$ はもとの基底計画に対する第 j 活動の限界価値で，正ではない。まとめると

$$\bar{p}_3'=p_3-\boldsymbol{u}_{\overline{B}}\boldsymbol{a}_3\leqq 0 \quad (23_1)$$

$$\bar{p}_j'=\bar{p}_j-\bar{p}_3't_{1j}\leqq 0\quad(j\neq 3) \quad (23_2)$$

(23₂)においてとくに $j=s$ の場合

$$\bar{p}_s'=\bar{p}_s-\bar{p}_3't_{1s}=0 \quad (23_3)$$

が成立しなければならない。

(23₃)で $\bar{p}_s\leqq 0$ でなければならないが，それには

$$t_{1s}<0 \quad (24)$$

であればよい。また(23₂)において $\bar{p}_j<0$, $\bar{p}_3'\leqq 0$ ゆえ，$t_{1j}\geqq 0$ であるような j については(23₂)は自然に成立する。問題は $t_{1j}<0$ であるような j であるが，このよう

な j については(23₃)，(24)を用いると

$$\bar{p}_j'=\left(\frac{\bar{p}_j}{t_{1j}}-\frac{\bar{p}_s}{t_{1s}}\right)t_{1j}\leqq 0$$

が成立することが要求される。したがって

$t_{1j}<0$ となるすべての j について

$$\frac{\bar{p}_j}{t_{1j}}-\frac{\bar{p}_s}{t_{1s}}\geqq 0 \qquad (25)$$

が成立すればよい。けっきょく s は次の2条件をみたすように選べばよいことになる

$$t_{1s}<0$$

$$\frac{\bar{p}_s}{t_{1s}}=\min_{t_{1j}<0}\left(\frac{\bar{p}_j}{t_{1j}}\right) \qquad (26)$$

上の例では

$$t_{11}=-\frac{15}{32}<0\ ,\ t_{14}=\frac{9}{32}>0$$

なので $s=1$ と選ばれる。したがって

$$\boldsymbol{B}=[\boldsymbol{a}_1,\ \boldsymbol{a}_2]=\begin{bmatrix}3 & 4\\ 12 & 6\end{bmatrix}$$

$$\boldsymbol{B}^{-1}=\begin{pmatrix}-\frac{1}{5} & \frac{2}{15}\\ \frac{2}{5} & -\frac{1}{10}\end{pmatrix}$$

$$\bar{\boldsymbol{d}}=\begin{bmatrix}400\\ 600\end{bmatrix}$$

$$\boldsymbol{X}_B=\bar{\boldsymbol{B}}^{-1}\bar{\boldsymbol{d}}=\begin{pmatrix}-\frac{1}{5} & \frac{2}{15}\\ \frac{2}{5} & -\frac{1}{10}\end{pmatrix}\begin{bmatrix}400\\ 600\end{bmatrix}=\begin{bmatrix}0\\ 100\end{bmatrix}$$

$$z=\boldsymbol{p}_B\boldsymbol{X}_B=[3,\ 2]\begin{bmatrix}0\\ 100\end{bmatrix}=200$$

$$\boldsymbol{u}_{\bar{B}}=\boldsymbol{p}_B\boldsymbol{B}^{-1}=[3,\ 2]\begin{pmatrix}-\frac{1}{5} & \frac{2}{15}\\ \frac{2}{5} & -\frac{1}{10}\end{pmatrix}=\left[\frac{1}{5},\ \frac{1}{5}\right]$$

$$\bar{p}_3'=p_3-\boldsymbol{u}_B\boldsymbol{a}_3=4-\left[\frac{1}{5},\ \frac{1}{5}\right]\begin{bmatrix}16\\ 8\end{bmatrix}=4-\frac{24}{5}=-\frac{4}{5}<0$$

$$\bar{p}_4'=p_4-\boldsymbol{u}_B\boldsymbol{a}_4=3-\left[\frac{1}{5},\ \frac{1}{5}\right]\begin{bmatrix}9\\ 9\end{bmatrix}=3-\frac{18}{5}=-\frac{3}{5}<0$$

以上で新しい基底計画が最適計画であることが確かめられた。ここでさらに d_2 を 600 より増やすことを考える。$\bar{\boldsymbol{B}}=(\boldsymbol{a}_1,\ \boldsymbol{a}_2)$ を改めて $\boldsymbol{B}$ と書き

$$\boldsymbol{d}=\begin{bmatrix}400\\ 600+\theta\end{bmatrix}$$

とおく。

$$\boldsymbol{X}_B=\boldsymbol{B}^{-1}\boldsymbol{d}=\begin{bmatrix}0\\ 100\end{bmatrix}+\begin{pmatrix}\frac{2}{15}\\ -\frac{1}{10}\end{pmatrix}\theta\geqq 0$$

が成立しなければならないから，θ の許容範囲は

$$0\leqq\theta\leqq 1000 \qquad (27)$$

となる。θ が 1000 を越えると x_2 が負になるから，$\boldsymbol{a}_2$ を基底から追い出さねばならない。しかし

$$t_{23}=[0,\ 1]\boldsymbol{B}^{-1}\boldsymbol{a}_3=\left[\frac{2}{5}\ ,-\frac{1}{10}\right]\begin{bmatrix}16\\ 8\end{bmatrix}=\frac{28}{5}>0$$

$$t_{24}=[0,\ 1]\boldsymbol{B}^{-1}\boldsymbol{a}_4=\left[\frac{2}{5}\ ,-\frac{1}{10}\right]\begin{bmatrix}9\\ 9\end{bmatrix}=\frac{27}{10}>0$$

となり，代わりに取り入れるべき活動がなくなる。事実 θ が1000を越えると実行可能な計画は存在しなくなる。つまり原料が1600トンを越えるともはや使いきれなくなるのである。

最初の最適計画から出発して原料を減少させていくほうも同様な方法で追跡することができるが，原料が 200 トン以下になるとこんどは設備容量が多すぎて，これを使いきる計画は存在しなくなる。結果を図2にまとめておこう。

以上で利益係数と総投入量をパラメタとして最適計画の分析を行なった。順序としては次に投入産出係数をパラメタとする場合を論ずべきであるが，この問題はここで扱うには繁雑すぎるので割愛させてもらう。とくに興味のある読者は Dantzig の前掲書を参照されたい。

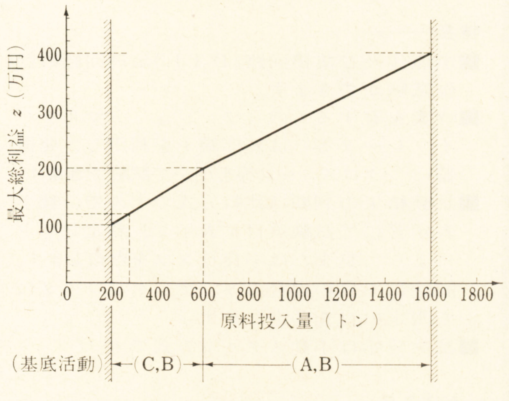


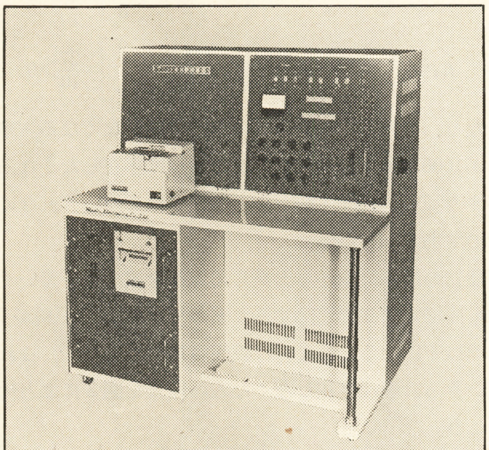
図 2

3. 双対単体法と双対定理

総投入量をパラメタとして変化させながら最適計画を追跡していく過程で，基底活動を入れかえる一つの方法を説明した。この方法は**双対単体法**とよばれており，いろいろな面で単体法とうらはらの関係にある。とくに双対単体法に対してふつうの単体法を**主単体法**とよぶことがある。ここで両者の特徴を比較対照してみよう。

扱う線形計画問題は

パッケージ自動試験装置



PKG-101型

プリント基板に装着された半導体論理回路に対する“動作”の良否判定を行うもので、入出力装置にテープリーダ、デジタルコーダを用いてあるため、操作は非常に簡単です。

特長

■半導体及び集積回路(DTL)論理回路の動作試験ができます。
■時素を有するもの（ワンショットマルチバイブレータ等）は外部端子を利用し、発振器・シンクロスコープ等を用いて試験できます。
■試験結果の判断は比較方式ですので、数字表示管にて試験項目を表示し、判定コードにより、良または不良ランプを点灯します。同時に不良の場合のみ、デジタルコーダにて測定項目を印字します。
■試験開始は手動スタート方式です。

製造品目

半導体IC自動選別器／各種カーブトレーサ／SCR測定器／その他半導体IC測定器／デジタル回路／符号変換器／その他論理回路応用機器／ナノセカンドパルス発生器／高出力パルス発生器／データパルス発生器／プログラムパルス発生器／高周波電源装置／クライストロン電源／進行波管電源／トランジスタ安定化電源／微少インダクタンス測定器／広帯域増巾器／高圧バルボン／半導体IC及び電子管製造設備／その他総合測定装置設計製作

港通信機株式會社
Minato Electronics Co., Ltd.

本社・工場□横浜市港北区南山田4105 ☎045 (591) 3611代
東京連絡所□東京都港区芝西久保八幡町9 ☎03 (431) 2733-8923
高崎工場□群馬県高崎市和田多中町162 ☎0273 (23) 2424

$$\max z = \boldsymbol{px} \mid \boldsymbol{Ax} = \boldsymbol{d},\ \boldsymbol{x} \geqq \boldsymbol{0}$$

とする。

	主単体法	双対単体法
初期基底行列 $\boldsymbol{B}$	$\boldsymbol{Bx}_B = \boldsymbol{d}$ $\boldsymbol{x}_B \geqq \boldsymbol{0}$	$\boldsymbol{u}_B\boldsymbol{B} = \boldsymbol{p}_B$ $\boldsymbol{u}_B\boldsymbol{A} \geqq \boldsymbol{p}$
基底ベクトルの入れかえ	取り入れるベクトル $\boldsymbol{a}_s$ を決定する： $\boldsymbol{u}_B\boldsymbol{a}_s < p_s$ 追出すベクトル $\boldsymbol{a}_{i_r}$ を決定する： $t_{rs} > 0$ & $\dfrac{x_{ir}}{t_{rs}} = \min\limits_{t_{ks}>0} \dfrac{x_{ik}}{t_{ks}}$ ただし $\boldsymbol{t}_s = (t_{ks}) = \boldsymbol{B}^{-1}\boldsymbol{a}_s$	追出すベクトル $\boldsymbol{a}_{i_r}$ を決定する： $x_{i_r} < 0$; $\boldsymbol{x}_B = (x_{i_1}, x_{i_2}, \cdots, x_{i_r}, \cdots x_{i_m})$ 取り入れるベクトル $\boldsymbol{a}_s$ を決定する： $t_{rs} < 0$ $\dfrac{\bar{p}_s}{t_{rs}} = \min\limits_{t_{rj}<0} \dfrac{\bar{p}_j}{t_{rj}}$ ただし $\bar{p}_j = p_j - \boldsymbol{u}_B\boldsymbol{a}_j$
目的関数値	非減少	非増加
最終判定	最適解の判定：すべての $\boldsymbol{a}_j$ について $\boldsymbol{u}_B\boldsymbol{a}_j \geqq p_j$ すなわち $\boldsymbol{u}_B\boldsymbol{A} \geqq \boldsymbol{p}$ 無限解の判定：すべての k について $t_{ks} \leqq 0$	最適解の判定：すべての k について $x_{i_k} \geqq 0$ すなわち $\boldsymbol{X}_B \geqq \boldsymbol{0}$ 不能の判定：すべての j について $t_{rj} \geqq 0$

$\boldsymbol{Bx}_B = \boldsymbol{d}$, $\boldsymbol{x}_B \geqq \boldsymbol{0}$ をみたす基底 $\boldsymbol{B}$ を主実行可能基底とよび，$\boldsymbol{u}_B\boldsymbol{B} = \boldsymbol{p}_B$, $\boldsymbol{u}_B\boldsymbol{A} \geqq \boldsymbol{p}$ をみたす基底 $\boldsymbol{B}$ を双対実行可能基底とよぶことにすれば，主単体法はある主実行可能基底から出発して目的関数値が増大するように基底ベクトル入れかえが行なわれ，その基底が双対実行可能基底に到達したとき最適基底となる。また双対単体法はある双対実行可能基底から出発して目的関数値が減少するように基底ベクトル入れかえが行なわれ，その基底が主実行可能基底に到達したとき最適基底となる。

主単体法の物理的意味は投入資源の収支関係を保ちながら利益最大化をたどる方法であり，双対単体法は投入資源の帰属価格を総投入資源の評価価値を最小化するように修正していく方法である。そして主単体法で得られる最大利益と双対単体法で得られる最小評価価値が一致することになる。この事実は次の**双対定理**を裏づけることになる。

主問題：

$$\max z = \boldsymbol{px} \mid \boldsymbol{Ax} = \boldsymbol{d},\ \boldsymbol{x} \geqq \boldsymbol{0} \qquad (28)$$

双対問題：

$$\min z' = \boldsymbol{ud} \mid \boldsymbol{uA} \geqq \boldsymbol{p} \qquad (29)$$

において，それぞれの条件式をみたす $\boldsymbol{x}$ と $\boldsymbol{u}$ をそれぞ

れの問題の実行可能解という。

双対定理　主問題と双対問題がともに実行可能解をもつならば，両問題にはそれぞれ最適解 $\boldsymbol{x}^*$，$\boldsymbol{u}^*$ が存在し

$$\boldsymbol{px}^*=\boldsymbol{u}^*\boldsymbol{d} \qquad (30)$$

が成立する。

〔証明〕任意の実行可能解 $\boldsymbol{x}$ と $\boldsymbol{u}$ について一般に

$$\boldsymbol{px}\leqq\boldsymbol{ud} \qquad (31)$$

が成立する。なぜならば実行可能解 $\boldsymbol{x}$，$\boldsymbol{u}$ については

$$\boldsymbol{uA}-\boldsymbol{p}\geqq\boldsymbol{0},\ \boldsymbol{x}\geqq\boldsymbol{0}$$

が成立していなければならないから

$$(\boldsymbol{uA}-\boldsymbol{p})\boldsymbol{x}\geqq 0$$

あるいは

$$\boldsymbol{uAx}-\boldsymbol{px}\geqq 0$$

となる。また実行可能解 $\boldsymbol{x}$ については

$$\boldsymbol{Ax}=\boldsymbol{d}$$

が成立していなければならないので

$$\boldsymbol{ud}-\boldsymbol{px}\geqq 0$$

すなわち(31)が証明された。

この事実は主問題のあらゆる実行可能解 $\boldsymbol{x}$ に対する目的関数の値 $z=\boldsymbol{px}$ が上に有界であり，また双対問題のあらゆる実行可能解に対する目的関数の値 $z'=\boldsymbol{ud}$ が下に有界であることを示している。したがって単体法によって必ず最適基底解 $\boldsymbol{x}_B^*$ が得られると同時にこの基底 $\boldsymbol{B}^*$ に対応して定まる $\boldsymbol{u}_B^*=\boldsymbol{p}_B^*\boldsymbol{B}^{*-1}$ は双対実行可能解であって

$$\boldsymbol{p}_B\boldsymbol{x}_B^*=\boldsymbol{u}_B^*\boldsymbol{d}$$

をみたす。そこで $\boldsymbol{u}_B^*$ が実は双対問題の最適解になっている。

次回は「線形計画法の応用」の予定

〔巻頭言〕彌永昌吉

〔連　載〕電子計算機とは・プログラミング入門・PL/I・論理設計・市場調査の話(完)・線形計画法

書評，ニュース　ドクタービット　単語帖，雑記帖　マンガ・ルポ

bit 8月号 予告

アレフ・ゼロ　coffee break　I/O シリーズ　読者の広場など

呉　清源〔ずいひつ〕

イリアック物語②　加賀美鉄雄・赤木昭夫〔よみもの〕

システムズ・エンジニアリング　中西義郎〔境界領域〕

OS入門②　浦　昭二・原田賢一〔準 連 載〕

B5判・88ページ・定価 260 円

共立出版株式会社

練習問題解答（出題は21ページ）

1

プログラム			説　明
	READ		カードを1枚読み込み
	INI	BANGO, 2, 5	品目番号5桁 BANGOへ
	INI	KOSU, 8, 3	個数　3桁　KOSU へ
	INI	TANKA, 12, 4	単価　4桁　TANKA へ
	LAD	TANKA	
	MLT	KOSU	
	STR	URIAG	売り上げ高 (URIAG)＝単価×個数
	OUTI	BANGO, 7	品目番号　7 桁 出力バッファへ
	OUTI	KOSU, 6	個数　5 桁 出力バッファへ
	OUTI	TANKA, 6	単価　6 桁 出力バッファへ
	OUTI	URIAG, 10	売り上げ高　10桁 出力バッファへ
	PRINT	*	頁きりかえ後，出力バッファ内容をプリント
	HALT		停止
BANGO		RS 1	品目番号
KOSU		RS 1	個数
TANKA		RS 1	単価
URIAG		RS 1	売り上げ高
	START		実行開始

2

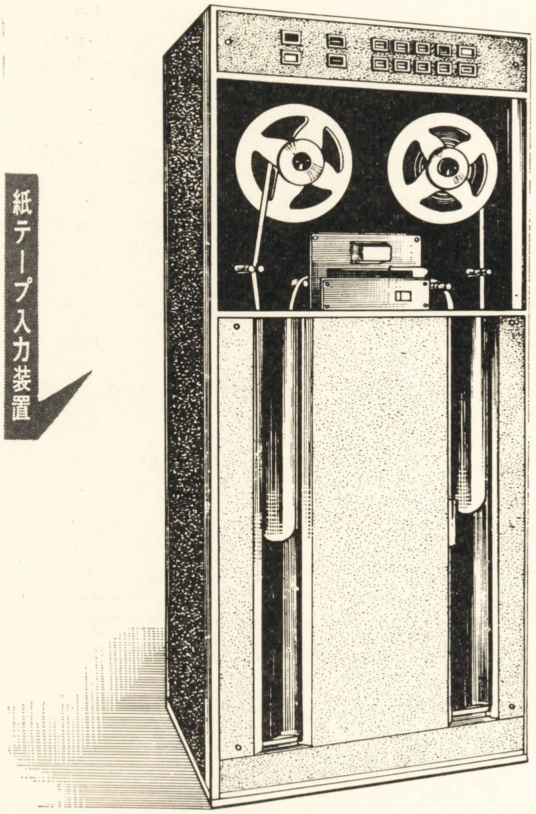
プログラム			説　明
	LADC	B	
	MLT	B	
	STR	W	$-b^2\to$W
	LADC	C	
	MLT	D	$-cd\to$AR
	ADD	W	
	ADD	A	
	STR	X	$a-(b^2+cd)\to$X
	OUTH	A＝	
	OUTI	A, 6	A＝±×××××　のプリント
	PRINT	*	
	OUTH	B＝	
	OUTI	B, 6	B＝±×××××　のプリント
	PRINT	1	
	OUTH	C＝	
	OUTI	C, 6	C＝±×××××　のプリント
	PRINT	1	
	OUTH	D＝	
	OUTI	D, 6	D＝±×××××　のプリント
	PRINT	1	
	OUTH	X＝	
	OUTI	X, 6	X＝±×××××　のプリント
	PRINT	2	
	HALT		停止
A		61527	a
B		74	b
C		−308	c
D		157	d
X		RS 1	x
W		RS 1	
	START		

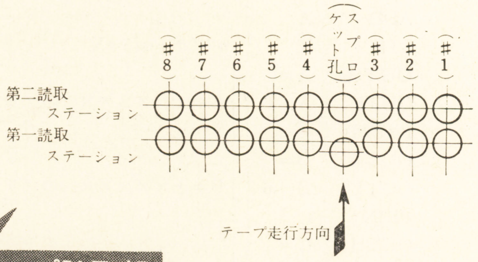
図解Ｉ／Ｏシリーズ③

イラスト／原　英夫

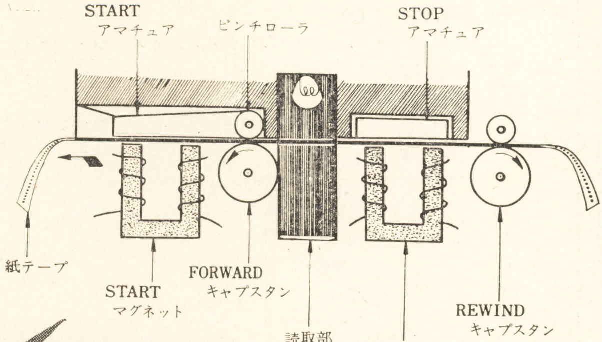
紙テープ入力装置

紙テープ入力装置





テープ読取部



テープ送り機構

紙テープ入力装置（Paper Tape Reader），紙テープ出力装置（Paper Tape Punch）は安価なので，コンピュータの入出力機器およびデータ伝送用の端末機器として多くの分野に用いられている。

次に，紙テープ規格，紙テープコード，紙テープ入力装置，紙テープ出力装置の構造について簡単に説明する。

1.　紙テープ規格

紙テープには幅および単位数により次の種類がある。

①紙テープ幅　25.4mm……8単位または7単位
②紙テープ幅　22.2mm……6単位
③紙テープ幅　17.5mm……5単位

図（右頁中央）には8単位紙テープ規格を示す。

2.　紙テープコード

紙テープ上の情報の表現は，1フレーム上にせん孔された孔の組合せにより行なわれる。

紙テープ入力装置からの情報の入力は，孔のある場所は1ビットとして，孔のない場所は0ビットとしてコンピュータの主記憶に格納される。

紙テープ上の各フレームの情報は主記憶の各番地に対応しているわけであるが，紙テープコードとコンピュータの内部コードは一般に異なっているので，データの処理を行なう前にプログラムでコード変換をする必要がある。図（右頁下）に紙テープコードの一例を示す。

3.　紙テープ入力装置，出力装置の構造

テープ送り機構

テープ送りの方法には大きくわけると，フリクションを利用して連続的に送る方法と，スプロケット（SP）孔を利用してピンにより1ピッチずつ逐一的に送る方法とがある。

フリクションを利用したテープ送りの方法は現在，高速の紙テープ入力装置に多く使用されているもので，その機構を図（左下）に示す。

図において FORWARD, REWIND キャプスタンは常に定速で回転している。読取指令が与えられると，START マグネットが励磁されピンチローラが FORWARD キャプスタンに圧着され，紙テープは急激にフィードされる。読取部における読取が終了するとSTOP マグネットが励磁され，瞬時にテープは停止する。紙テープの走行停止は，200～1,000フレーム/秒で高速に走行されているものを1フレームピッチ（2.54mm）以内に確実に停止するように制御される。

紙テープ出力装置は，パンチ動作を行なうためテープの間けつ送りが必要となる。従って，ラチェット機構によりフィードホイールを1フレームピッチずつ回転させ，スプロケット孔を1フレームピッチ分ずつテープ送りを行なう。

テープ読取部

テープ読取り方式には機械式のものと光電方式のもの

紙テープ出力装置

との２種類があるが，高速入力装置では一般的に読取りの信頼性が高い光電方式が多く採用されている。

なお，読取り情報の信頼性を高めるため，読取りステーションをテープ１ピッチの間隔で二重に配置し，二重読取りを行なっているものもある。図（左頁中央）にその例を示す。

テープせん孔機構部

テープへの情報のせん孔はテープの間けつ送りに同期したパンチピンの上下運動によって行なわれる。図（右下）にせん孔機構の一例を示すが，パンチピンの上下運動は，パンチベイルの定常的な上下運動をトグルリンケージによって伝達することによって行なわれる。すなわち，情報をせん孔する場合は，コードマグネットが励磁され，トグルリンケージはせん孔するためのパンチピンに直結され，パンチベイルが下になったとき，せん孔される。

チェック方式

紙テープ機器における主なチェックとしては次のものがあげられる。

①パリティ・チェック：　１フレームの情報孔の数が必ず偶数個または奇数個になっているかどうかチェックするもの。

②二重読取り：　紙テープ入力装置において同じデータを二度読み照合するもの。

③フィード・チェック：　紙テープのフィードが正しく行なわれているかどうかチェックするもの。

④エンドチェック：　紙テープ出力装置において紙テープの終端を検出するもの。

4. 性能例

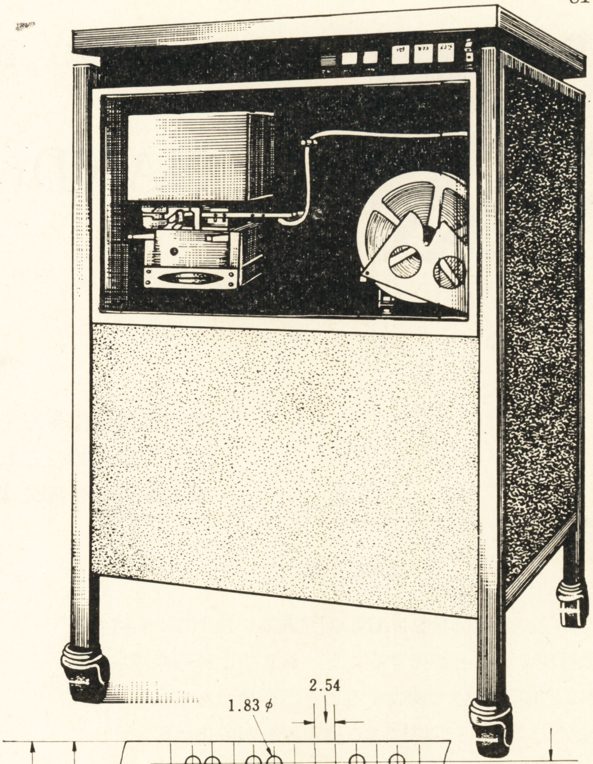
紙テープ入力装置性能例

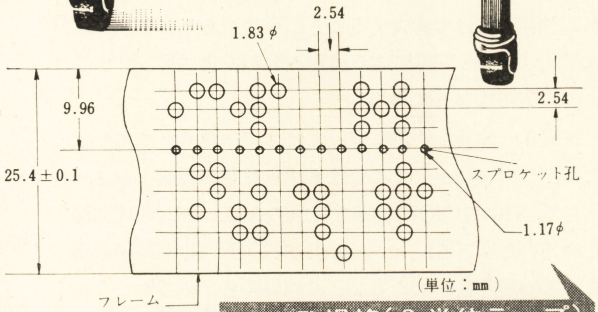
読取方式　　光電式
読取速度　　1,000フレーム/秒
紙テープ単位数　５～８単位
チェック方式　　パリティ・チェック，二重読取り

紙テープ出力装置性能例

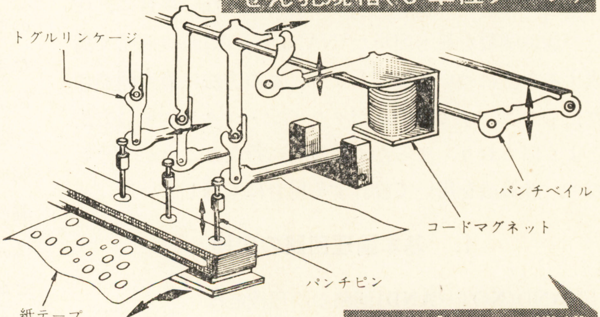
せん孔方式　　ダイパンチ
せん孔速度　　110フレーム/秒
紙テープ単位数　５～８単位
チェック　　パリティ・チェック，エンド・チェック

（日本電気KK　千秋　勝）

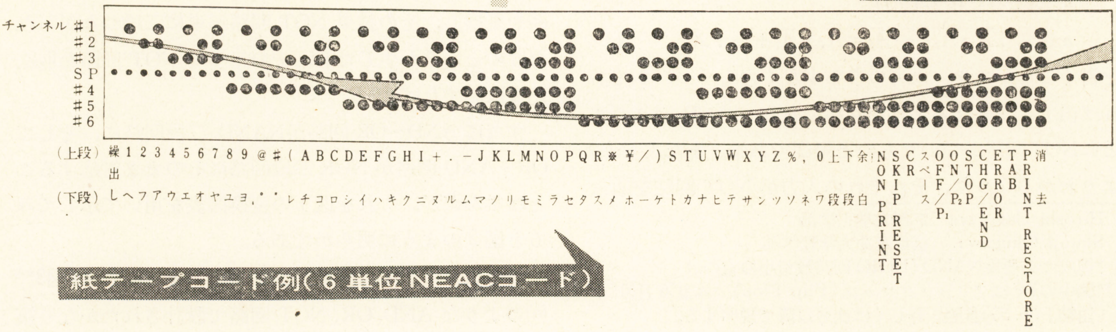




せん孔規格（８単位テープ）



テープせん孔機構



紙テープコード例（６単位NEACコード）

連載　論理設計⑤

NAND(NOR)回路

萩原　宏・奥川　俊二

集積回路化に伴って重要になってきたNAND回路とNOR回路についてまとめ，これらの回路で任意の論理関数をなるべく簡単に構成する方法を考えよう

はじめに

任意の論理関数をNAND回路あるいはNOR回路のみで構成することができること，およびこれらの基本回路がIC(集積回路)で構成することが容易であることから，NAND (NOR) 回路による論理設計がIC化に伴ってひじょうに重要になってきた。

NAND (NOR) 回路についてはすでに前号までにしばしばふれてきたが，ここではまずそれらの機能をおさらいして，NAND(NOR)回路と従来のAND, OR, NOT回路との間の変換方法，与えられたNAND(NOR)回路網の構成する論理関数の求め方などについてまとめておこう。

そして変数の補元が入力として得られない場合に，任意の論理関数を最も簡単な*NAND(NOR)三段回路で構成する方法について説明しよう。この方法は最近開発されたもので，すでに学んだクワイン・マクラスキの方法とよく似ており，まったく機械的に行なうことができるので，計算機を用いて設計する場合にも適している。

1.　基本論理回路の変換

NAND(**NOT－AND**)回路の論理機能は

2変数：$\overline{A \cdot B} = \overline{A} + \overline{B}$　(1)

n変数：$\overline{A_1 \cdot A_2 \cdot \cdots \cdot A_n} = \overline{A}_1 + \overline{A}_2 + \cdots + \overline{A}_n$　(2)

と書くことができ，入力のうち少なくとも一つが0であれば出力が1となるような回路であり，AND回路のあとにNOT回路がついたものと等価である。また式(1), (2)よりわかるように，それぞれの入力がNOT回路を通ったのちOR回路の入力となる回路とも等価である。

またNOR(**NOT－OR**)回路の論理機能は

2変数：$\overline{A + B} = \overline{A} \cdot \overline{B}$　(3)

n変数：$\overline{A_1 + A_2 + \cdots + A_n} = \overline{A}_1 \cdot \overline{A}_2 \cdot \cdots \cdot \overline{A}_n$　(4)

であり，入力のすべてが0のときだけ出力が1となるような回路であって，OR回路のあとにNOT回路のついた回路と，そしてまた式(3), (4)よりわかるようにそれぞれの入力がNOT回路を経てAND回路につながれる回路と等価である。なお，一入力のNAND回路およびNOR回路はNOT回路と等価である。

以上のNAND(NOR)回路とAND, OR, NOT回路との間の変換方法を**図1**にまとめて示しておこう。これらの変換は式(1), (3)より明らかであろう。図1では簡単のため2変数の場合について示してあるが，もちろん3変数以上の場合も同様である。

図1より，AND⇌NAND, OR⇌NORの変換では出力側にNOT回路を，AND⇌NOR, OR⇌NAND では入力側にNOT回路を，そしてAND⇌OR, NAND⇌NORでは入力および出力側ともにNOT回路をつければよいことがわかる。したがってどのような基本回路で構成した論理回路においても，上に示した関係を用いて各基本回路を変換し，その結果NOT回路が単に二段つながっただけのところが生じたら，その二つのNOT回路を取除いて直結すればよい。

すでにAND－OR回路がNAND二段回路に，そしてOR－AND回路がNOR二段回路にそのまま対応することは学んだが，これを図1の変換図表を用いて示してみると**図2**のように明らかである。

このように図1の変換図表を用いると，たとえば**図3****(a)のようなAND, OR, NOT回路で設計されたかなり複

Hiroshi Hagiwara　京都大学工学部
Shunji Okugawa　京都産業大学理学部

*構成に必要なNAND(NOR)回路数最小のもの。

**図中のFFはフリップフロップ(Flip-Flop)　本誌6月号「単語帖」(p. 55)参照。詳しくは次号以降で説明する。

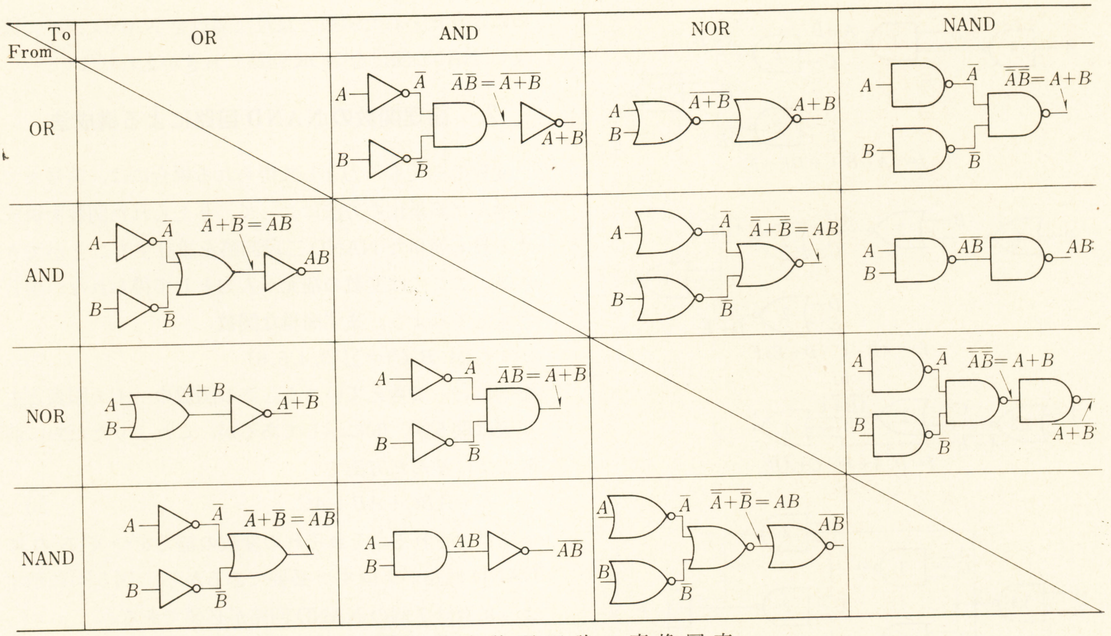


図1　基本論理回路の変換図表

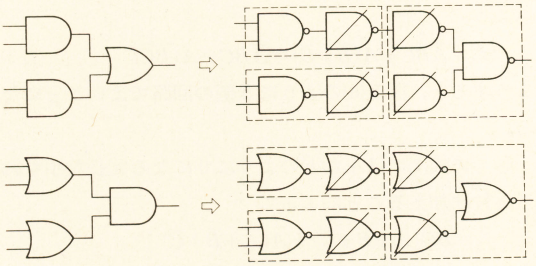


図2　AND-OR (OR−AND) 回路から NAND (NOR) 二段回路への変換

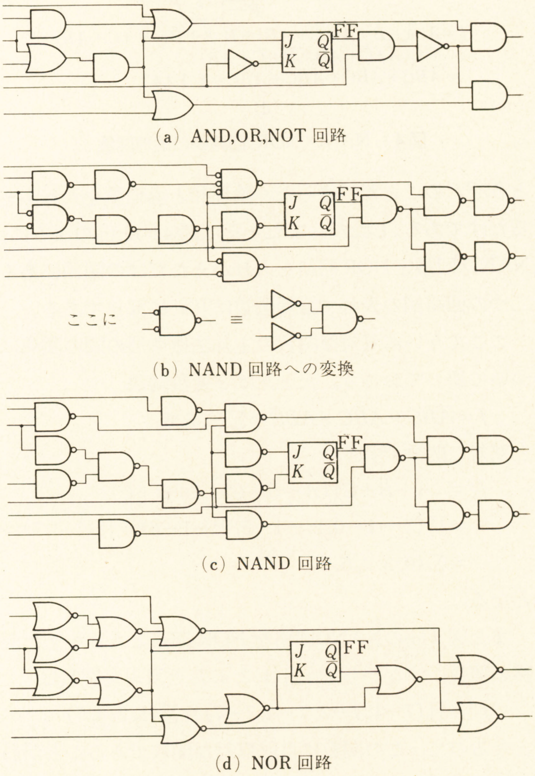


図3　AND, OR, NOT 回路から NAND (NOR) 回路への変換例

雑な回路も(b), (c), (d)に示すようにNAND(NOR) 回路に容易に変換することができる。

2.　NAND(NOR) 回路網と論理関数

与えられたNAND(NOR) 回路網が構成する論理関数を求めることを考えよう。**図4***よりわかるように，NAND (NOR) 回路網が構成する関数を変数とその補元に論理積および論理和演算をほどこした論理式の形で表わすには次のようにすればよい（かっこ内は NOR の場合）。

(1)　出力側から数えて奇数段の回路は論理和(積)に，偶数段の回路は論理積(和)

(2)　偶数段と奇数段の2種の回路の入力となるような出力をもつ回路は，奇数段と偶数段の回路の二つにわけて考え，それぞれを論理和(積)，論理積(和) に対応させる。

(3)　奇数段にはいる変数はすべて補元にする。

このようにNAND回路の場合とNOR回路の場合とは，

*各回路につけてある番号は出力側から数えて何段目になるかを示す。

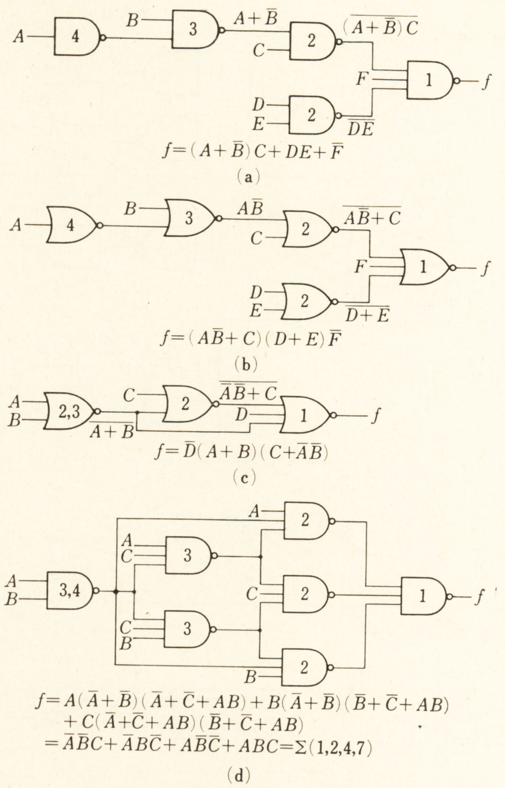


図4　NAND(NOR)回路網の論理関数

とうぜんのことだがまったく論理和と論理積を入れかえるだけである。したがってたとえば図4(a)と(b)のように，NAND回路とNOR回路を入れかえただけの関係にある二つの回路網の表わす論理関数は互いに双対である。

ここで互いに双対な関数の主加法標準形の間の関係について調べておこう。たとえば3変数関数

$$f=\bar{A}\bar{B}\bar{C}+A\bar{B}C+AB\bar{C}=\sum(0,5,6)$$

と双対な関数 f^d は*

$$f^d=(\bar{A}+\bar{B}+\bar{C})(A+\bar{B}+C)(A+B+\bar{C})$$
$$=\prod(2^3-1-0,2^3-1-5,2^3-1-6)=\prod(1,2,7)$$
$$=\sum(0,3,4,5,6)$$

となる。

また図4(d)のNAND回路をNOR回路と入れかえた回路の関数，すなわち $f=\sum(1,2,4,7)$ と双対な関数は

$$f^d=\prod(7-1,7-2,7-4,7-7)=\sum(1,2,4,7)$$

となるので，この関数は自己双対な関数であることがわかる。

以上のようにNAND回路網とNOR回路網とは双対の関係にあるので，以下ではおもに任意の関数のNAND回路による構成法を説明するが，NOR回路で構成するためには与えられた関数と双対な関数をNAND回路で構成し，NAND回路をNOR回路におきかえればよい。

2.　論理関数のNAND回路による構成法

変数の補元も入力として得られる場合には，前号までに学んだ簡単化の方法によって，与えられた関数を構成する最も簡単なNAND二段回路を設計することができるが，ここでは変数の補元が入力として得られない場合を考えてみよう。まず簡単な関数

$$f_1(A,B,C)=\sum(3,4,5,6) \qquad (5)$$

を例にとって考えていこう。この関数のベイチ図表と主項表は**図5**(a)，(b)に示してあるが，これよりただちに前号までに学んだ方法で

$$f_1=\bar{A}BC+A\bar{B}+A\bar{C} \qquad (6)$$

と簡単化される。それぞれの変数の補元をつくるのにNAND回路が1個ずつ必要であるから，図5(c)に示すように合計7個のNAND回路が必要である。

そこでこの式(6)を次のように分配則を用いて変形してみよう。

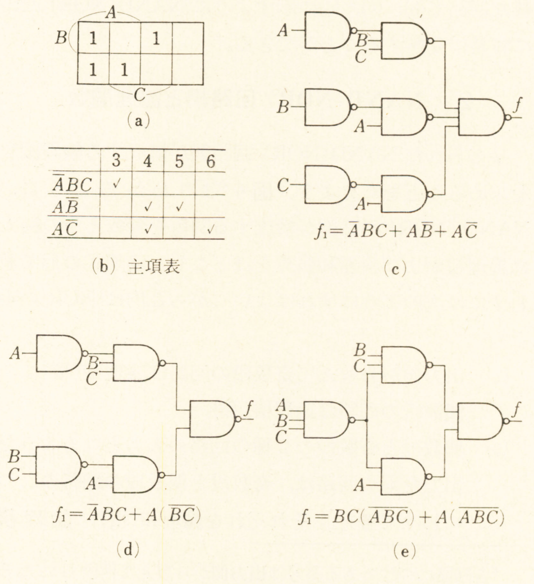
$$f_1=\bar{A}BC+A(\bar{B}+\bar{C})=\bar{A}BC+A(\overline{BC}) \qquad (7)$$

こうすると，図5(d)のように5個の回路でよいことになる。

さらに式(7)に $A\bar{A}$ などのように0になる冗長な項をわざと付け加えることによって

$$f_1=BC(\bar{A}+\bar{B}+\bar{C})+A(\bar{A}+\bar{B}+\bar{C})$$
$$=BC(\overline{ABC})+A(\overline{ABC}) \qquad (8)$$

のように変形してみると，(e)のように合計4個のNAND



	3	4	5	6
$\bar{A}BC$	✓			
$A\bar{B}$		✓	✓	
$A\bar{C}$		✓		✓

図5　f_1 の NAND 回路による構成

*f^d の d は dual（双対）の頭文字をとったもの。

回路で構成できることになる。

このように，いままで学んだ方法で得た最も簡単なNAND二段回路に変数の補元を作るためのNAND回路を三段目に付け加えた回路よりも，さらに簡単な三段回路が存在するわけである。そして一般に変数の補元が入力として得られない場合でも，**NAND 三段回路***で任意の論理関数を実現することができるのである。

さて，ここで，このようなNAND三段回路とそれに対応する論理式との間の関係を一般的に考察しておこう。たとえば，**図6**の回路ではその表わす論理式は

$$f=A(\overline{BC})+DE(\overline{FG})+(\overline{BC})(\overline{FG})+BF \qquad (9)$$

となる。これと式(6)，(7)，(8)からわかるように，各項は一般に

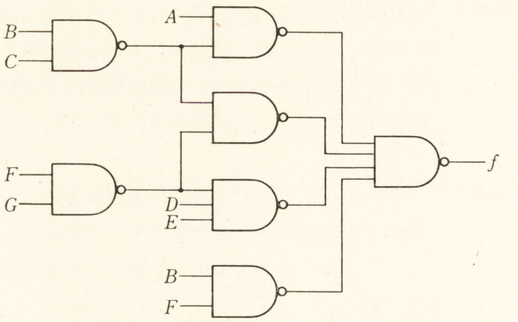
$$H\cdot\overline{T_1}\cdot\overline{T_2}\cdots\cdots\overline{T_m} \qquad (10)$$

なる形をしている。ここに$H, T_1, T_2, \cdots, T_m$はすべて変数の積の形である。そしてこのHの部分をH部，$\overline{T_1}, \overline{T_2}, \cdots, \overline{T_m}$をそれぞれ$\boldsymbol{T}$**因子**，$\overline{T_1}\cdot\overline{T_2}\cdots\cdot\overline{T_m}$を$\boldsymbol{T}$**部**と名づけておこう（例よりもわかるように$T$部，$H$部はないこともある）。

そうすると式(9)の形の論理式の中の相異なるT因子の数が第三段のNAND回路の数（この例では2）となり，項の数が第二段のNAND回路の数（この例では4）となることは図6より容易に理解できるだろう。そしてこの二つの数の和に1（第一段のNAND回路数）を加えたものが必要なNAND回路の総数となるから，式(9)のような形をした論理式の中で，**相異なる$\boldsymbol{T}$因子の数と項数の和が最小**になるようなものを見出せばよいことになる。

4. NAND 三段回路の設計法

前節の例のような簡単な関数の場合には，分配則を用いたり冗長な項を付け加えたりして，式(8)のような論理



$f=A(\overline{BC})+DE(\overline{FG})+(\overline{BC})(\overline{FG})+BF$

図6 NAND 三段回路

*このような回路を，TANT 回路(Three-level And-Not Network with True Inputs) とよぶことがある。

■最新刊■
経営のための
数理計画法
ゲルシゴルン著／坂本実訳 A5判¥900
かつては経験の蓄積と直観に基礎をおいた国や企業の運営は、今や厳密な科学的根拠にもとづく計画によって裏づけられねばならない。電子計算機の開発と発展は数学によるこの裏づけを可能とした。本書は産業経済の担い手達に経済活動の精確な定量的研究方法を与えようとするもので、最適需要の把握、資本投下や情報伝達の最適化など企業運営に必須の数学的根拠を平易に解説する。

科学者のための FORTRAN
田中一、小野周、国井利泰著 A5判¥980
著者たちが毎日の計算機利用の経験をもとにFORTRANによるプログラミングを入門から実際の計算が行なえるまで、国産各機種の共通部分を中心に多数例で解説。初心者には良き入門書となり、熟練者には座右のハンドブックとなる好テキスト。

誤差論の基礎
アゲキャン著／本田、幾志訳 A5判¥750
実験値、観測値を取り扱う際、そのデータからいかにして真の値に近いものを推定するかが問題となる。本書はまず確率・統計の基礎概念から述べ、誤差論の根拠を明らかにし、従来のあいまいな誤差論から統計的推定の問題へと論じた好入門書。

線形代数と線形計画法
ソロドフニコフ著 坂本 実訳 A5／¥900
例題・練習問題を豊富におりこみ，できるだけ平易な理論と方法を用いて簡潔にしかも数学的な厳密さを少しも損うことなく解説した好入門書。

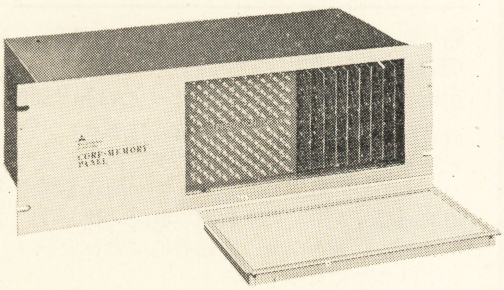
最適過程の数学的理論
ポントリャーギン他著 関根智明訳 A5 ¥2000
最大値原理の理論と応用を詳細かつ厳密に展開した世界的名著。訳注を詳しく付して読者の理解を助けている。

最適制御の数学的方法
ボルチャンスキー著 坂本 実訳 A5 ¥1600
著者独自の簡明な数学的手法によって，難解な最大値理論を平明に説いた好入門書。

東京都千代田区 神田神保町1-30 総合図書 振替東京 97477 電話 294−6700

三菱
コアメモリ

あなたは自社のシステムに
コアメモリの導入を
決意なさいましたか？
コアメモリは
記憶装置の主流です



●コンピューターに
●計測制御に
●コントロールシステムに
●小形計算器用レジスタに

三菱電機株式会社

お問合せは
電材営業部
東京都千代田区丸ノ内2−12 〒100
電話 東京 03〈212〉6111〈大代表〉
大阪商品営業所応用電機課
大阪市北区堂島北町8−1 〒530
電話 大阪 06〈344〉1231〈大代表〉
名古屋商品営業所応用電機課
名古屋市中村区広井町3−88 大名古屋ビル 〒450
電話 名古屋 052〈561〉5311〈大代表〉
相模製作所
相模原市宮下1−1−57 〒229
電話 相模原 0427〈72〉5131〈大代表〉

式を導くことができた。しかし一般には，このようにブール代数の公式をうまく用いて式を変形する方法はたいへん困難であり，また得られた結果が最も簡単かどうかわからない場合が多い。

この節では最近**ギンペル**によって見出された一般的な方法* をわかりやすさに重点をおいて説明していこう。すでに学んだクワイン・マクラスキの方法では，まず最簡形式を構成する項の候補者として主項を求め，次に主項表を作成してすべての簡約形式を表わす主項関数を求め，これにより最簡形式を見出した。これと同様にギンペルの方法も，まず主項に相当する候補者であるところの**主許容項****を求め，主項表と主項関数に相当する表と論理式を求めて最簡形式***を導くのである。

1. 主許容項の求め方

前節式(8)からもわかるように，一般に式(10)のような項には，たとえば

$$\left.\begin{aligned} &AB(\overline{CD})=AB(\overline{ACD})=AB(\overline{BCD})=AB(\overline{ABCD})\\ &A(\overline{BC})(\overline{BD})=A(\overline{ABC})(\overline{BD})=A(\overline{BC})(\overline{ABD})\\ &\qquad=A(\overline{ABC})(\overline{ABD}) \end{aligned}\right\}\quad(11)$$

のように，H部がk個の変数よりなりT因子がl個あれば，2^{kl}個の等価な項が可能であるが，T部からH部の変数をすべて除いた$AB(\overline{CD})$，$A(\overline{BC})(\overline{BD})$のような項をそれらの代表として取り扱うことにする。そして関数fに含まれるこのような代表の項で，これよりどのT因子を取除いても，もはや関数fに含まれなくなるようなものをfの**主許容項**とよぶことにする。したがって主項もすべて主許容項である。なお，主許容項のT因子のうち少なくとも一つが複数個の変数より成り立っているときこれを複合主許容項（**複合項**）と名づけることにする。

さて以上のような主許容項を求める手順は

手順1　主項を求め，主項表を作る。

手順2　主項表で二つ以上の主項に含まれるような最小項に注目し，その最小項を含むすべての主項の論理和とその最小項のH部との論理積を作り式(10)の形の項を求める。これが複合項にならないときはすてる。

手順3　手順2で得た複合項と同じH部をもち，この複合項に含まれるような複合項をすべて求める。

のようになり，これらの手順1から手順3で求まった項が主許容項である。

---

*J. F. Gimpel: The Minimization of TANT Networks, IEEE Trans. on EC, vol. EC-16, No. 1, Feb. 1967

**prime permissible implicant

***必要なNAND回路数最小のもの。

表 1

		$\overline{BC}$	$\overline{BD}$
$\bar{B}$	a	√	√
$\bar{C}$	b	√	
$\overline{BC}$	c	√	
$\bar{D}$	d		√
$\overline{BD}$	e		√

前節で例に用いた式(5)で与えられる関数の場合には，図5(b)の主項表よりわかるように最小項4だけが複数個の主項に含まれ，手順2によって

$$A(A\bar{B}+A\bar{C})=A(\overline{BC})$$

なる複合項が求まる。手順3では新たな複合項は求まらないので，この関数の主許容項は$BC\bar{A}$, $A\bar{B}$, $A\bar{C}$, $A(\overline{BC})$の計4個である。

ここで手順3をもう少し説明しておこう。手順2でたとえば$A(\overline{BCD})$なる複合項が求まったとすると，これに含まれるH部がAの複合項は，$A(\overline{BC})$, $A(\overline{BD})$, $A(\overline{CD})$の3個がある。また手順2でたとえば$A(\overline{BC})(\overline{BD})$が求まったとすれば，$A\bar{C}(\overline{BD})$, $A(\overline{BC})\bar{D}$なる2個の複合項が求まる（そのほかは$A\bar{B}(\overline{BD})=A(\overline{BC})\bar{B}=A\bar{B}$となり複合項でない）。この手順を一般的に行なうためには**表1**のような表を作り主項関数と同様の論理式を求めて展開すれば，元の複合項のT部に含まれるようなすべてのT部が求まるのでこの中から複合のものを選び出せばよい。表1ではたとえばT因子$\overline{BC}$に含まれるようなT因子は，$\bar{B}$, $\bar{C}$, $\overline{BC}$の3通りがあり，$\overline{BD}$に含まれるのは$\bar{B}$, $\bar{D}$, $\overline{BD}$の3通りゆえ

$$(a+b+c)(a+d+e)=a+(b+c)(d+e)$$
$$=a+bd+be+cd+ce \qquad (12)$$

によって，このなかからbeとcdに対応する$\bar{C}(\overline{BD})$と$(\overline{BC})\bar{D}$なる複合T因子が求まる。

2. 主許容項の選び方

次に以上の手順で求まった主許容項とそのT部をどう選んだらよいかを考えよう。クワイン・マクラスキの主項表に対応する**表2**のような主許容項表を用いると便利である。この表の左上部はそれぞれの主許容項がどの最小項を含むかを示すものであって，主項表とほとんど同じものである。

表2　f_1の主許容項表

			③	4	5	6				
主許容項	$BC\bar{A}$	ⓐ	√				*			
	$A\bar{B}$	b		√	√			*		
	$A\bar{C}$	c		√		√			*	
	$A(\overline{BC})$	d		√	√	√				*
T因子	$\bar{A}$	ⓔ					√			
	$\overline{AB}$	f					√	√		
	$\overline{AC}$	g					√		√	
	$\overline{ABC}$	h					√			√
	$\bar{B}$	ⓘ						√		
	$\bar{C}$	ⓙ							√	
	$\overline{BC}$	ⓚ								√

最簡形式を求めるためには，項数と相異なるT因子の数の和が最小になるような，主許容項とそのT因子の組合せを選べばよい。たとえば表2の第1行の主許容項$BC\bar{A}$は一つのT因子をもつが，このT因子は$\bar{A}$, $\overline{AB}$, $\overline{AC}$, $\overline{ABC}$なる4個の等価な形が可能である。したがって第1行には一つ＊印をつけて一つの列をとり上記の4個のT因子の行をもうけ，この列の交点の欄に√印をつける。もし主許容項が複数個のT因子をもてば，それぞれの因子について＊印をつけた一つずつの列を用意すればよい。

次に第2行の$A\bar{B}$を考えると，一つのT因子をもちこの因子は$\bar{B}$のほかに$\overline{AB}$なる等価な形が可能である。すでに$\overline{AB}$なるT因子の行はもうけられてあるので対応する欄に√印をつけ，$\bar{B}$なるT因子の行はまだないので新たにもうけて同様に√印をつける。以下すべての主許容項についてこのような操作を行なって表2が完成される。

このようにして求まった表を簡単にすることを考えよう。たとえば行iのように，その行が√印をもつすべての列に√印をもつ行fのような行があるときには，前者が後者に含まれるので前者は表から省くことができる。表2ではこのようにして丸印をつけた行e, i, j, kが不要である。

さてこの簡単化された表を使って，クワイン・マクラスキの方法における主項関数に相当する論理式を求めることを考えよう。最小項の列に対しては，いままでの主項関数とまったく同様にして

$$p=a(b+c+d)(b+d)(c+d)=a(d+bc) \qquad (13)$$

なる論理式を考えればよい。

そしてT因子をどう選べばよいかを示す＊印のついた列に対しては

$$q=(\bar{a}+f+g+h)(\bar{b}+f)(\bar{c}+g)(\bar{d}+h) \qquad (14)$$

なる論理式を考えればよい。この式(14)の一番目の因子はもし行a，すなわち主許容項$BC\bar{A}$を選べば，行f, g, hのうちのどれか，すなわちT因子として$\overline{AB}$, $\overline{AC}$, $\overline{ABC}$のうちのどれかを選ぶべきことを示している。そしてまた，もし行aを選ばなければ，$\bar{a}=1$となってその因子は1になり考慮しなくてよいことを示している。

以上のpとqの論理積を積の和の形に展開して最も補元でない変数の少ない項を見出せばよい。この例では

$$p\cdot q=a(f+g+h)(\bar{b}+f)(\bar{c}+g)(dh+bc\bar{d})$$

表3 f_2 の主許容項表

				⓪	①	3	④	⑤	6	9	11	13										
主許容項	主項	$\overline{A}\overline{C}$	ⓐ	√	√		√	√					*	*								
		$D\overline{B}$	b		√	√				√	√				*							
		$D\overline{C}$	c		√			√		√		√				*						
		$B\overline{A}\overline{D}$	d				√		√								*	*				
	複合項	$D(\overline{BC})$	e		√	√		√		√	√	√							*			
		$B\overline{A}(\overline{CD})$	f				√	√	√											*	*	
		$AD(\overline{BC})$	g							√	√	√										*
T因子		$\overline{A}$	h										√				√			√		
		$\overline{C}$	i											√		√						
		$\overline{B}$													√							
		$\overline{BD}$	j												√			√				
		$\overline{CD}$	k													√					√	
		$\overline{AB}$															√			√		
		$\overline{D}$																√				
		$\overline{BC}$																	√			√
		$\overline{BCD}$	l																√		√	√
		$\overline{ABC}$																				√
		$\overline{ABCD}$																				√

$$=adh\bar{b}\bar{c}+\cdots\cdots \quad (15)$$

となり，主許容項として ad に対応する $BC\overline{A}$ と $A(\overline{BC})$ とを，そして T 因子は h に対応する $\overline{ABC}$ を選べばよいことがわかるから，けっきょく

$$f_1=BC(\overline{ABC})+A(\overline{ABC})$$

を得る。これは前節の式(8)と同じものであり，回路図は図5(e)に示されている。

理解を深めるためにもう少し例をあげておこう。

例1 関数

$$f_2(A,B,C,D)=\sum(0,1,3,4,5,6,9,11,13) \quad (16)$$

を考えよう。主項は表3の左上の主項表の部分からわかるように $\overline{A}\overline{C}, D\overline{B}, D\overline{C}, B\overline{A}\overline{D}$ の5個あり，最小項1, 4, 5, 9がそれぞれ複数個の主項に含まれる。複合項を求めるため前述の手順2を行なえば，それぞれの最小項に対して

$$\left.\begin{aligned}
\text{最小項}1: &\quad D(\overline{A}\overline{C}+D\overline{B}+D\overline{C})=D(\overline{BC})\\
4: &\quad B(\overline{A}\overline{C}+B\overline{A}\overline{D})=B\overline{A}(\overline{CD})\\
5: &\quad BD(\overline{A}\overline{C}+D\overline{C})=BD\overline{C}\\
9: &\quad AD(D\overline{B}+D\overline{C})=AD(\overline{BC})
\end{aligned}\right\} \quad (17)$$

となり複合項が三つ求まる。そしてこれらの複合項に含まれるような複合項はないので主許容項を求める手順は終了する。そして f_1 に対して表2を作成したのと同様にして表を完成させ，他の T 因子の行に含まれてしまうような T 因子の行は除いて残った行に $a, b, \cdots, l$ と記入する。この表によって主許容項 $\overline{A}\overline{C}$ は最小項0を含む唯一の項，すなわち必須項であることがわかるから，a に丸印をつけこの項に含まれる最小項0, 1, 4, 5も丸で囲む。

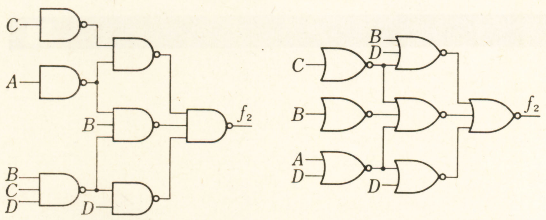
主項関数と同様の論理式を書き下すと

$$\begin{aligned}
p&=a(b+e)(d+f)(b+c+e+g)(b+e+g)(c+e+g)\\
&=a(b+e)(d+f)(c+e+g)\\
&=a(d+f)(e+bc+bg)
\end{aligned} \quad (18)$$

さらに右側の＊印を含む列に対して

$$\begin{aligned}
q&=(\bar{a}+h)(\bar{a}+i)\cdot(\bar{b}+j)(\bar{c}+i+k)\cdot(\bar{d}+h)(\bar{d}+j)\\
&\quad(\bar{e}+l)\cdot(\bar{f}+h)(\bar{f}+k+l)\cdot(\bar{g}+l)\\
&=(\bar{a}+hi)(\bar{b}+j)(\bar{c}+i+k)(\bar{d}+hj)(\bar{e}+l)\\
&\quad(\bar{f}+hk+hl)(\bar{g}+l)
\end{aligned} \quad (19)$$

この二つの論理式の積 $p\cdot q$ から最も少ない補元でない変数よりなる項として $aef\cdot hil\cdot\bar{b}\bar{c}\bar{d}\bar{g}$ が求まる。したがって主許容項として aef に対応する $\overline{A}\overline{C}, D(\overline{BC}), B\overline{A}(\overline{CD})$ が求まり，それらの T 因子として hil に対応する $\overline{A}, \overline{C}, \overline{BCD}$ が求まるから，図7(a)に示すようにNAND回路7個で関数 f_2 を構成できる。



$f_2=\bar{A}\bar{C}+D(\overline{BCD})+B\bar{A}(\overline{BCD})$　(a)

$f_2^d=D(\overline{AD})+BD\bar{C}+\bar{B}\bar{C}(\overline{AD})$　(b)

図7　関数 f_2 のNAND（NOR）三段回路による構成

5.　NOR三段回路の設計法

すでに2節で説明したようにNAND回路とNOR回路の双対性を利用することによって，与えられた関数と双対な関数を前節の方法によって NAND 三段回路で構成し，NAND回路をNOR 回路に入れかえればよい。ギンペルの方法のもう一つの例としての意味からも，最後に例1で扱った関数 f_2 を NOR 三段回路で構成することを考えよう。

*可能なかぎり簡潔にクワイン・マクラスキの方法と対照させて方法の骨組を説明したつもりである。細部については前にあげた文献を参照していただきたい。

例2　f_2 と双対な関数 f_2^d は

$$f_2^d=\prod(15,14,12,11,10,9,6,4,2)$$
$$=\sum(0,1,3,5,7,8,13) \qquad (20)$$

である。これに対する主許容項表は**表4**のようになり，

$$p=a(c+e)(b+f) \qquad (21)$$
$$q=(\bar{a}+g+h)(\bar{b}+i)(\bar{c}+ijk)(\bar{d}+gij)$$
$$(\bar{e}+hij)(\bar{f}+l) \qquad (22)$$

となるから，最も簡単な主許容項およびその T 因子の組合せを示す項として，$abe\cdot hij\cdot\bar{c}\bar{d}\bar{f}$ が求まる。したがって

$$f_2^d=D(\overline{AD})+BD\bar{C}+\bar{B}\bar{C}(\overline{AD}) \qquad (23)$$

となるから図7(b)に示すように関数 f_2 は NOR 回路7個で構成することができる。

以上説明したギンペルの方法はかなりめんどうなようであるが，操作はクワイン・マクラスキの方法とよく似ていてまったく機械的なため，計算機を用いて設計するのは容易である*。

次回は「順序回路(1)」の予定

表4　f_2^d の主許容項表

				0	①	③	⑤	⑦	8	13												
主許容項	主項	$D\bar{A}$	ⓐ		√	√	√	√			*											
		$BD\bar{C}$	b				√			√		*										
		$\bar{B}\bar{C}\bar{D}$	c	√					√				*	*	*							
		$\bar{A}\bar{B}\bar{C}$	d	√	√											*	*	*				
	複合項	$\bar{B}\bar{C}(\overline{AD})$	e	√	√				√										*	*	*	
		$BD(\overline{AC})$	f				√	√		√												*
T因子		$\bar{A}$	g								√					√						
		$\overline{AD}$	h								√										√	
		$\bar{C}$	i									√		√				√		√		
		$\overline{BC}$										√										
		$\overline{CD}$										√										
		$\overline{BCD}$										√										
		$\bar{B}$	j										√				√		√			
		$\bar{D}$	k												√							
		$\overline{AC}$	l																			√
		$\overline{ABC}$																				√
		$\overline{ACD}$																				√
		$\overline{ABCD}$																				√

■コンピュータ紹介 ■

NEAC-シリーズ2200

1. 概　要

NEAC-シリーズ2200は，コンピュータの国産納入実績No.1を続けるNEC（日本電気）が長年の豊富な経験による通信技術をもとに完成したコンピュータ群で，バッチ処理はもちろん，オンライン・リアルタイム処理，タイムシェアリング用に好適な配慮が多数なされている。機種名の NEAC（ニアック）は Nippon Electric Automatic Computer の略である。

本シリーズには九つのモデルがあり，ソフトウェアはすべてのモデルに共通使用できる，いわゆる One Machine Concept により一貫している。したがって，ソフトウェアを変更することなく，仕事量の増加にともなって，より大きなシステムに拡張できるようになっている。（写真はモデル150）

2. ハードウェア

1. 演算速度（単位はμ秒）

演算＼モデル	50	100	150	200	250	300	400	500*	700*
加減算（十進5桁±5桁）	63	63	55	48	36	34.5	26	6	2.2
比較（十進5桁）	51	51	42.5	38	31.5	28.5	23	6	1.2
分岐（条件付）	18	18	15	14	13.5	10.5	10	4.5	1.0
転送（20字）	138	138	115	96	73.5	72	51	15	3.6
浮動小数点加減算**	—	—	—	—	19.5	19.5	15	4.5	0.8

注：モデル 50，100，150：2字アドレス・モード
　　モデル200，250，300：3字アドレス・モード
　　モデル400，500，700：4字アドレス・モード
　　の場合を示した。

* 命令を先回り制御により先取りされている場合を示し，オペランドは1語に含まれている場合を示す（ただし，転送は3語に相当）。

** 48ビットで演算を行なった場合の値

2. 主記憶容量

＼モデル	50	100	150	200	250	300	400	500	700
最大容量（K字）	16	32	32	65	131	131	262	524	2,097
サイクル・タイム（μs/字）	2	2	2	2	1.5	1.5	1.0	1.5/8字	1.5/8字

外部記憶装置には各種のディスクパック装置，ディスクファイル装置，磁気ドラム装置，磁気テープ装置が用意されている。

3. 入出力装置

代表的な入出力装置として，光学文字読取装置，光学マーク読取装置，紙テープ入出力装置，カード読取せん孔装置，高速製表印字装置，マン-マシン・コミュニケーションを容易にする操作卓，入出力タイプライタ，NEACデータスコープ，ディジタル・プロッタがある。これらの装置は本シリーズのすべてのモデルについて，システムのレベルに応じ任意に組み合わせることができる。

4. データ通信機器

多数・多種類の通信回線を制御する通信制御装置，および遠隔地に設置される汎用なデータステーション，NEACデータスコープ，NEAC TYPER，エーゼント装置などの各種の端末装置がある。

5. 切換装置

周辺制御切換装置（任意の2台から4台までの中央処理装置間で，周辺制御部と周辺装置を切換える），磁気テープ切換装置，通信線切換部など。

3. ソフトウェア

1. EASYプログラミング・システム

(1) EASY BILL：伝票発行業務を主とし，販売管理・在庫管理の即時処理を同時に行なう。

(2) EASY PRO ：一括処理に伴うデータの入力・分類・照合・帳票作成などを簡単な項目指定で行なう。

(3) EASY COBOL：日本語コボルで，規模の大きい複雑な業務に使う。

〔最小機器構成〕

テープ・サブシステム：中央処理装置——主記憶8K字，磁気テープ装置4台，紙テープ入力装置1台，高速製表印字装置1台

ディスク・サブシステム：中央処理装置——主記憶8K字，ディスクパック装置1台，紙テープ入力装置1台，入出力タイプライタ1台

2. オペレーティング・システム（OS）

(1) オペレーティング・システム MODI MSR

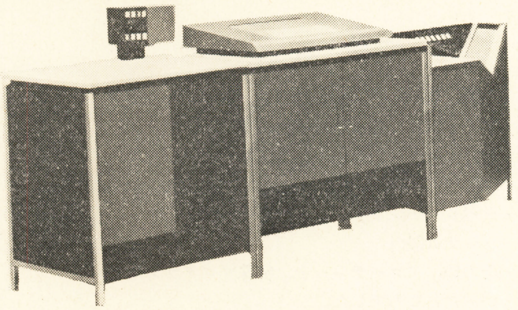
ランダム・アクセス・ファイル装置を中心とした小ないし中規模のシステムを効果的に運用するための OS。COBOL，FORTRAN のコンパイラ使用可能。

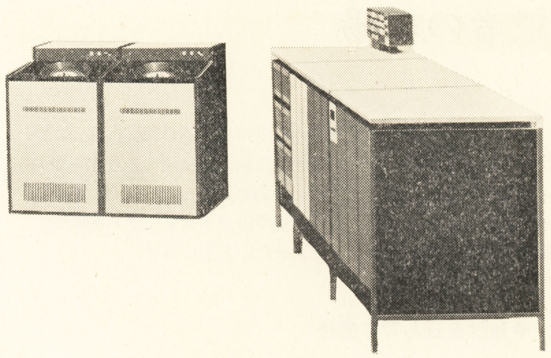
〔最小機器構成〕

中央処理装置——主記憶12K字，ディスクパック1台，カード読取装置または紙テープ入力装置1台，高速製表印字装置1台

(2) オペレーティング・システム MODI TR

磁気テープ装置を中心とした小ないし中規模のシステムのための OS。COBOL，FORTRAN，ALGOL の各コ





ンパイラ使用可能。

〔最小機器構成〕

中央処理装置——主記憶12K字，磁気テープ装置３台，カード読取装置または紙テープ入力装置１台，高速製表印字装置１台

(3) オペレーティング・システム MODI EXTENDED
前記二OSの機能をすべて含み，中規模以上のシステムに多重プログラム処理，オンライン・リアルタイム処理の機能を追加するOS。

(4) オペレーティング・システムMOD Ⅲ
おもにモデル400と500においてバッチ処理の形式で使用される。とくに機能的に独立した二つのスタックト・ジョブ，および一つのスタックト・ジョブと三つのデータ交換ジョブの同時処理が行なえる。COBOL，FORTRAN，ALGOL，PL/I，RPG 使用可能。

〔最小機器構成〕

中央処理装置——主記憶98K字，ディスクパック装置１台，磁気テープ装置３台，カード読取装置１台，高速製表印字装置１台，操作卓１台，その他付加機構など

(5) オペレーティング・システムMOD Ⅳ
MOD Ⅲのバッチ処理能力をさらに拡大し，リアルタイム処理能力をもたせたもので，TSSにも適したOS。

(6) オペレーティング・システム MOD Ⅳ EXTENDED
MOD Ⅳと互換性をもち，その機能をさらに拡張して，モデル700のシングル・プロセッサ・システムで使用される。

(7) オペレーティング・システム Ⅶ
MOD Ⅳ EXTENDEDと互換性をもち，その機能をさらに拡張し，モデル700のマルチ・プロセッサ・システムで使用される。とくに，このシステムは高速磁気ドラム装置または磁気ディスク装置を中心にした構成で，システムのオーバーヘッドを減少することができる。

4. アプリケーション・システム

1. 経営科学

効率的な企業経営のために，組織的な分析・計画・管理の諸情報を提供する。

(1) 科学データ処理：数値計算，統計計算
(2) 統計的分析：市場調査，販売予測，品質管理，人事管理
(3) シミュレーション：最も成功の可能性のあるプランを選出する。たとえば，設備投資や予算配分のシミュレーションなど。
(4) 線形計画：生産計画，輸送計画，混合計画，人員配置計画などの最適配分問題を解く。
(5) ネットワーク分析：大きなプロジェクトに含まれる諸活動を，ネットワークで表現し，プロジェクトの進行管理と時間，費用，資源などのコントロールを行なう。
(6) 計量経済分析および需要予測：経済，産業，企業構造の分析・予測および政策の評価を行なう。

2. 情報検索

大量の情報の収集，整理，検索を行なう。

3. 生産・在庫管理システム

とくに製造業を主体としたシステムで，統合経営情報管理システムを完成できる。

4. その他

構造設計計算：平面骨組架構の応力，変形解析など

■ひとくち解説

バッチ処理(batch processing) 処理の対象となるデータをひとまとめにし，一回の機械操作で処理する方式。一括処理ともいう。リアルタイム処理の対語

オンライン(on-line) コンピュータの中央演算処理装置が直接，端末装置や周辺装置に結合している状態。オフラインの対語

リアルタイム処理(real-time processing) データの発生と同時に，そのデータを即時に処理して必要な結果が得られること。バッチ処理の対語

オペランド(operand) 演算される数のこと。演算数ともいう。たとえば，A＋BのAおよびBがオペランドである

ディスク(disk) 磁気ディスク装置または磁気ディスクそのものをさす。データは磁性材料を塗布した円盤(ディスク)に記憶される

ディスク・パック(disk pack) 取はずし可能な一組の磁気ディスクのこと。同一種の磁気ディスク装置の間で互換できる

アセンブラ(assembler) 記号語プログラムから機械語プログラムを作る翻訳プログラム。翻訳は１対１

コンパイラ(compiler) ソース・プログラムを機械語に翻訳するためのプログラム。ソース・プログラムの１ステートメントが数個の機械語となる

スタックト・ジョブ(stacked job) ある作業(ジョブ)から次のジョブに移る際に介在する手作業部分をプログラムによって排除し，自動的に連続処理すること

マルチ・プロセッサ(multi-processor) 一つのシステム中の，同時使用できる複数個の中央演算処理装置

読者の広場

愛読者カードとの対話

本誌5月号に挿入の愛読者カードが多数編集部宛に届いています。編集部では，これを今後の企画・編集面での貴重な資料とさせていただく考えですが，折角の読者との“対話”が一方通行に終わることのないよう，現段階でお答えできる範囲で，編集部の考えを明らかにしておきたいと思います。

◆bitを読んでの感想

［よいと思った記事］ これについては，ソフト，ハード，アプリケーションの各面に対する読者の関心度の違いからか，推奨された記事はかなりまちまちでしたが，なかでも，とくに多くの方から好評を得ているものとして，山本欣子先生の「プログラミング入門」と「単語帖」があげられると思います。このうち，「プログラミング入門」のほうは一年間の連載予定であり，「単語帖」は半永久的(？)に続く予定ですから，当分ご愛読いただけることと思います。

［内容の程度］ 小誌 **bit** の読者層は，学生・初中級コンピュータ要員・技術者・研究者・経営管理層と，たいへん幅の広いものになっています。したがって，すべてが入門的記事ばかりではなく，中級向きの，やや歯ごたえのある記事を盛りこんで，いろいろなレベルの要求に応えるようにしています。このことからも，ある程度予想されたことですが，内容の程度については，「ちょうどよい」が圧倒的に多いなかで，「高すぎる」という声も散見しました。編集部としては，これらのご意見を謙虚に拝聴し，今後とも“表現はやさしく，内容は高く”をモットーに努力していきたいと考えています。

◆今後とり上げてほしい記事

［ハードウェア］ 中央処理装置(CPU)については，CPU全般の解説ではなく，その中のとくに「回路素子」についての解説を希望する方が多数でした。小誌では，現在連載中の記事が完結次第，近く「素子の話」をひき続いて連載する予定です。

一方，周辺装置(peripheral)については，いわゆるI/O（入出力装置）と，OCRやグラフィック・ディスプレイについての記事が望まれているようです。小誌 **bit** の編集方針としても，ソフトウェアがとくにハードウェアに優先するとの立場をとってはいませんので，現在連載中の「図解I/Oシリーズ」の例などに見られるように，これらの周辺機器についても，随時解説していくつもりです。

［ソフトウェア］ 具体的に，いろいろなテーマが多数寄せられましたが，なかでも，希望の多かったものをあげますと

- FORTRAN, COBOLなどのコンパイラの解説
- アセンブラ，種々の機械語の紹介
- 統計，数値解析などの基礎数学
- OR，PERT，シミュレーションなどの応用プログラム
- オペレーティング・システム
- システム設計
- その他，フローチャート，特殊プログラムなど

といったところでした。

このうち，たとえばFORTRANなどについては，現在とくに陽わな形ではとり上げていませんが，小野・伊理両先生の「電子計算機とは」の中でのプログラミングにも，実質的な解説とともに使用されていますし，今後も，たえず多くの記事の中に当然とり入れられてくると思います。また，オペレーティング・システムについては，本月号から3回にわたって，浦・原田両先生に執筆をお願いしました。

さらに，コンピュータの基礎理論としての数学的な面，たとえば，統計学や数値解析などについても，当然のことながら，すでに連載企画のうちに組み入れてあります。

一般的にいって，これまでにあげたテーマのすべては，小誌 **bit** が標榜する“コンピュータ・サイエンス”のカテゴリーに属していますので，編集部としては，随時これらをとり上げていくことをお約束できると思います。これは，ハード，ソフト，アプリケーションのいずれの面についてもいえることです。

［アプリケーション］ これまでの小誌のスタイルからもおわかりのように，小誌では，アプリケーションの面をできるだけ視覚的に紹介する意味で，グラビアやマンガ・ルポの頁を設けてありますが，紙数の関係もあって，やや掘り下げ方が足りないのではないかと自戒しています。今後は，本文の紙数をグラビアの解説に当てるなどして，より充実したものにしていきたいと考えています。

以上，愛読者カードに寄せられた貴重な数々のご意見に，個々にお答えする代わりに，本誌上をかりて総括的に編集部の考えを述べさせていただきました。どうぞ今後とも末長く，小誌 **bit** をご愛読下さいますようお願い申上げます。

お断り 5月号に挿入しました愛読者カードの返送も，すでにピークを過ぎたように見えますので，ファイルの抽選の締切日を6月末日（当日消印有効）とさせていただきたいと思います。つきましては，まだ，ご投函いただいてない方は，できるだけ早目にご投函下さるようお願い致します。

なお，マンガ・ルポは都合により本月号は休載致しました。併せてご了承のほどお願い申上げます。

編集部

昭和44年7月1日発行
編集人 塩谷 茂
発行人 南條初五郎
印刷人 竹内勝之
印刷所 壮光舎印刷株式会社

© 1969 by Kyoritsu Shuppan Co.

—禁転載— 1969年 7月
Vol.1, No. 5
コンピュータ・サイエンス誌
bit

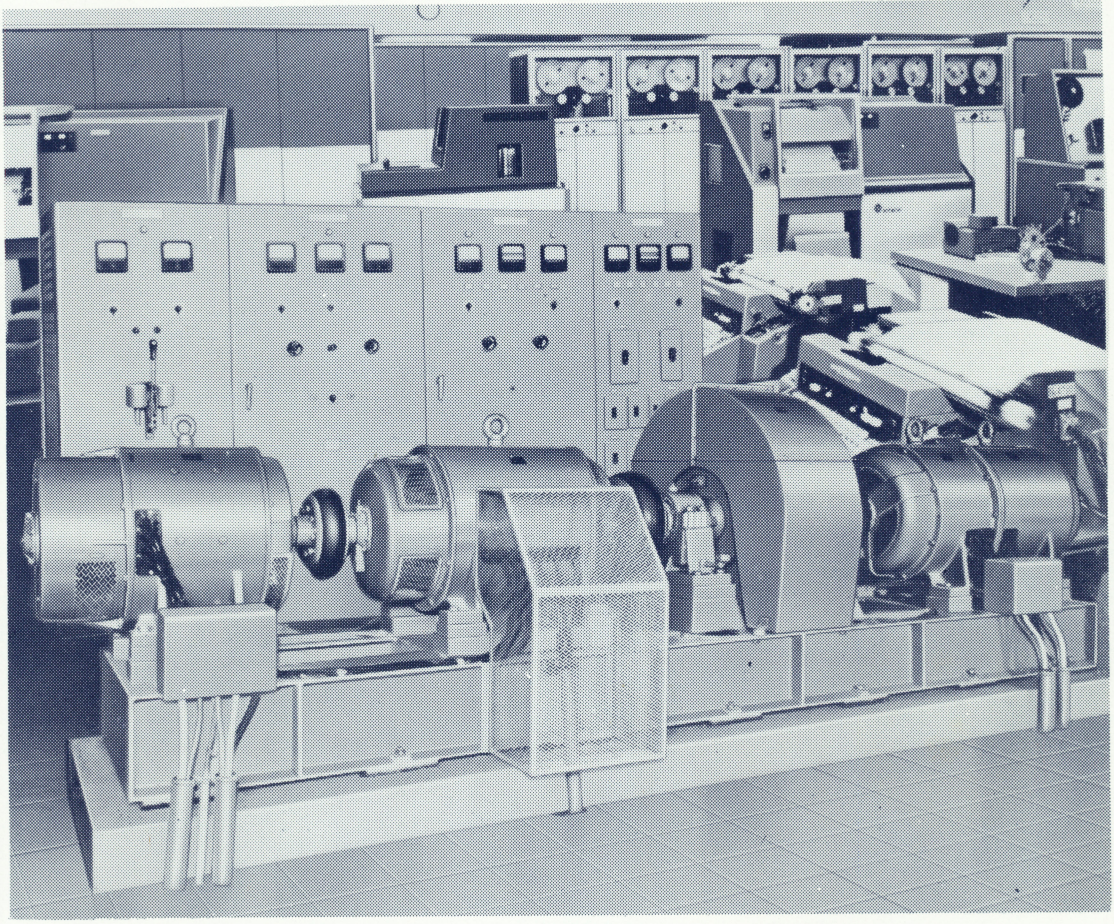
発行所 共立出版株式会社
東京都文京区小日向4—6—19 〒112
Tel. 03—947—2511
振替 東京 57035
定価 一部 260円（送料 18円）
年間購読料 3,120円

藤本喬雄

★高信頼性・高経済性と設置実績を誇る…………

情報化時代に活躍する！

山洋の電算機用電源装置



国税庁殿納入定周波定電圧電源装置

営業品目

- 定周波定電圧電源装置
- 電子計算機用関連機器
- ディーゼル機関発電装置
- 無停電電源装置
- 機械器具設置工事及電気配線工事
  (建設大臣登録（ヌ） 第4132号)

SANYO

山洋電気株式会社

本　社　東京都豊島区巣鴨6丁目1349番地
電話　東京(917)代表5151～9番
加入電信　(23)　476番
出張所　大阪市北区梅ケ枝町72番地（電子会館内）
工　場　東京・川口・上田

共立出版

情報の世界 コンピューターのすべて

<SABシリーズ1>

J・マッカーシー他著／南雲仁一訳　A5判・850円

本書は世界的著名雑誌サイエンティフィックアメリカン誌に連載された「情報―この驚くべき新技術の全貌」の全訳で、内容は電子計算機そのものの働き、その使用法から広汎な応用全般に及び、科学、工学、商業と経営、教育におけるその役割の現状と将来を平易に解説。

主な目的＝情報・計算機の論理と記憶・計算機の入出力・システム解析とプログラミング・計算機の時分割方式・データ伝送・科学における計算機の応用・工学における計算機の応用・社会組織における計算機の応用・教育における計算機の応用・情報の貯蔵と検索・人口知能

■続刊

材料の科学

<SABシリーズ2>　7月中旬刊

J・ザイマン他著／黒田　晴雄訳　A5判・850円

本書は「材料」の新しい科学・技術を平易に解説したもので、特にその発展を歴史的に捉え将来への展望を与えた

ノルム環

<共立講座現代の数学19>

和田　淳蔵著　A5判・1200円

ノルム環の一般論について述べるとともに、特に可換ノルム環、そのうちでも最近長足の進歩をなした新しい分野の一つである関数環の理論を紹介し、あわせてその意義と応用とについて論じた。

■既刊

6 アーベル群・代数群　本田欣哉・永田雅宜著……A5判・1000円

9 代数幾何学入門　中野茂男著……A5判・1000円

近　刊

自習 FORTRANテキスト

伊藤忠電子計算サービス株式会社編　　理学博士国井　利泰監修

B5判　約360ページ　予定価1800円 <7月下旬刊>

本書は、これから電子計算機(Computer)を使おうとする人達のためのフォートラン・プログラム(Fortran program)の入門書である。電子計算機を動かす一番やさしい「言葉」としてフォートランがあり、これを修得することによって誰でも電子計算機を使うことができるようになる。電子計算機に関して何も知らない人でも、ただ忠実にこの本を読んでいけば、必ずフォートラン・プログラムができるようになるのが、本書の目的である。

主な目次＝FORTRANについて・四則演算・大小関係と判断・くり返し制御・基本事項のまとめ・サブプログラム・タイプ宣言と各種計算・そのほかのステートメント・特殊なステートメント

計算機科学に職を求めよう

J.M.キャロル著　森口　繁一監訳・伊理　由美訳

B6判　約192ページ　予定価500円 <7月下旬刊>

本書は、計算機について何も知らない人に、計算機についての基本的な概念とその応用領域全般についての展望を与えるもので、アメリカでの現状を具体的に示しながら計算機関係の職業が将来性のある有利な職業で、いろいろなレベルの人々を広く招いているということを強調している。

共立出版案内—電気・電子工学書より

電気工学一般 改訂版
尾本義一・本城巌著／B6・256頁・¥580

解説電気理論
宮田房近著／A5・256頁・¥780

磁気
近角聡信編／A5・552頁・¥2500

演習 電磁気学
宮島竜興編／A5・170頁・¥480

磁気共鳴吸収・物質の電気的性質
永宮健夫編／A5・444頁・¥1800

演習 原子物理学
三輪光雄編／A5・220頁・¥580

演習 数理物理学・線型物理学
西山敏之他著／A5・240頁・¥580

マイクロウエーブ
霜田光一著／B6・184頁・¥450

電気磁気演習
尾本義一・小郷寛著／B6・260頁・¥550

電気回路理論
神保成吉・西山栄枝著／A5・324頁・¥950

交流理論
熊谷三郎・手塚慶一著／A5・304頁・¥950

交流理論演習
尾本義一・高井宏幸著／B6・180頁・¥400

回路網理論 I
尾崎弘・黒田一之著／B6・298頁・¥600

過度現象 増補版
宮田房近著／B6・288頁・¥600

解説自動制御
宮田房近著／B6・216頁・¥450

自動制御計算法 改装版
高橋安人編／A5・88頁・¥650

情報理論と通信方式
笠原芳郎著／A5・272頁・¥980

有線伝送学
笠原芳郎著／A5・288頁・¥950

新しい電子工学
岡部金治郎著／A5・192頁・¥600

新版 電子工学
岡部金治郎・松尾幸人著／A5・¥900

電子工学の基礎 I
W. G. ダウ著・森田清他訳／B6・¥700

電子工学の基礎 II
W. G. ダウ著・森田清他訳／B6・¥600

電子回路 I
川上正光著／B6・262頁・¥650

電子回路 II
川上正光著／B6・210頁・¥550

電子回路 III
川上正光著／B6・208頁・¥550

電子回路 IV
川上正光著／B6・256頁・¥650

電子回路 V
川上正光著／B6・278頁・¥700

電子回路演習 I
小郷寛著／B6・224頁・¥500

電子回路演習 II
小郷寛著／B6・190頁・¥580

エレクトロニクス回路演習
三戸左内他著／B6・244頁・¥580

ビジネスデータ処理機器
磯部孝他編／A5・240頁・¥500

交換工学
高柳晃著／A5・212頁・¥800

オペレーションズ・リサーチ
宮脇一男他著／A5・284頁・¥850

ORの一手法としての 線型計画法
野志功著／A5・210頁・¥500

工程・品質および原価の管理
山口襄他編／A5・296頁・¥900

市場調査と製品開発
茅野健他編／A5・242頁・¥900

解説 エレクトロニクスコース 全8巻

0. エレクトロニクスへのスタート
大島正光他編／A5・192頁・¥600

1. 電気と回路のはたらき
川上正光他編／A5・246頁・¥850

2. 真空管とトランジスタの話
古賀正三他編／A5・224頁・¥800

3. 電子回路のはたらき I
武藤義一他編／A5・226頁・¥800

4. 電子回路のはたらき II
武藤義一他編／A5・248頁・¥780

5. 自動制御とトランスジューサ
古賀正三他編／A5・218頁・¥700

6. 観測からデータ処理まで
川上正光他編／A5・218頁・¥750

7. 電子部品の使い方
大島正光他編／A5・272頁・¥780

基礎電気工学講座 全15巻より

1. 電気数学
尾本義一他編／A5・328頁・¥900

2. 原子物理学
武田栄一他編／A5・344頁・¥950

3. 電気物理学
鳩山道夫他編／A5・308頁・¥850

4. 電気用材料
宮部宏他編／A5・184頁・¥800

5. 電気磁気学
神保成吉著／A5・184頁・¥600

6. 電気回路論
安宅彦三郎著／A5・326頁・¥900

7. 伝送回路
滝保夫著／A5・274頁・¥800

8. 過渡現象論
熊谷三郎・尾崎弘著／A5・¥780

9. 電波工学
前田憲一著／A5・362頁・¥1000

11. 電気自動制御
神保成吉著／A5・256頁・¥750

12. 電子管とトランジスタ
西巻正郎著／A5・272頁・¥800

14. 磁気回路
茂木晃著／A5・298頁・¥850

15. 電磁機械 上・下
後藤良著／A5・¥各600

共立出版案内—数学書

数学小辞典
矢野健太郎編／B6・768頁・￥1900

共立 **数学公式**
泉信一他編／三六／改訂中

新しい基礎数学
J. N. フジイ著
柴垣和三雄・稲葉三男訳／A5・￥1200

一般教育 **新数学**
大学新数学教育研究会編／A5・￥450

大学教養 新課程 文科系 **一般数学**
稲葉三男著／A5・202頁・￥480

一般教養 **数学概論** 訂正増補
寺阪英孝他編／A5・288頁・￥500

新数学通論 I
福原満洲雄・稲葉三男著／A5・￥600

新数学通論 II
福原満洲雄・稲葉三男著／A5・￥700

大学課程 **数学**
大学数学教育研究会編／A5・￥600

社会科学のための **数学概説**
小林竜一著／A5・286頁・￥780

命題・集合・統計
黒田孝郎著／A5・264頁・￥680

新しい数学
J. G. ケメニー他著
矢野健太郎訳／A5・368頁・￥980

暮しのなかの数字 増補版
小林幹雄著／A5・258頁・￥680

禅と数学 増補版
坂井豊著／B6・252頁・￥600

アシモフ選集＝数学編 1
数の世界
矢野健太郎監修／B6・216頁・￥430

アシモフ選集＝数学編 2
量の世界
矢野健太郎監修／B6・222頁￥430

アシモフ選集＝数学編 4
速くてやさしい算数
矢野健太郎監修／B6・226頁・￥430

応用数学の基礎
П. И. ロマノフスキー著
久保忠雄訳／A5・232頁・￥750

工学のための **基礎数学**
斉藤渥・岡本正明著／A5・248頁・￥750

共立物理学講座 1
物理数学 1
堀淳一著／A5・276頁・￥880

ディジタル代数学
尾崎弘・樹下行三著／A5・378頁・￥1400

ラジオ・テレビに必要な **数学入門**
松平維石著／A5・288頁・￥780

新版 **集合論**
辻正次著／A5・192頁・￥550

ベクトル
稲葉三男著／A5・188頁・￥500

ベクトル解析入門
各務頼文・岩永照夫著／A5・280頁・￥780

ベクトル解析 新版
友近晋著／A5・168頁・￥480

基礎数学 **代数と幾何**
寺阪英孝他編／A5・258頁・￥500

理工科系 一般教育 **代数・幾何教科書**
占部実・光藤富士男編／A5・260頁・￥550

大学教養 新課程 理工科系 **代数学と幾何学**
小松醇郎・永田雅宜著／A5・230頁・￥500

代数学および幾何学
山内恭彦著／A5・256頁・￥900

詳解 **代数・幾何演習**
福田安蔵他編／A5・264頁・￥680

大学課程 **行列とベクトル**
大学数学教育研究会編／A5・238頁・￥550

詳解 **ベクトルと行列演習**
福田安蔵他編／A5・232頁・￥650

アシモフ選集＝数学編 3
代数の世界
矢野健太郎監修／B6・248頁・￥480

代数学講義 改訂新版
高木貞治著／A5・400頁・￥1600

初等整数論講義
高木貞治著／A5・改訂中

基礎数学 **微分と積分**
寺阪英孝他編／A5・248頁・￥500

微分積分学
小林幹雄著／A5・170頁・￥450

理工科系 一般教育 **微分・積分教科書**
占部実・佐々木右左編／A5・280頁・￥550

大学課程 **微分積分学概説**
大学数学教育研究会編/ A5・298頁・￥750

大学教養 新課程 理工科系 **微分学と積分学**
福原満洲雄・村松寿延著/A5・252頁・￥500

数学解析序説
佐藤徳意著／A5・248頁・￥750

微分積分学演習 上 改訂版
坂井英太郎著／A5・464頁・￥1200

微分積分学演習 下 改訂版
坂井英太郎著／A5・418頁・￥1200

位相解析
加藤敏夫著／A5・336頁・￥1400

複素関数概説
黒田正著／A5・206頁・￥680

函数概論 改訂版
竹内端三著／B6・284頁・￥600

解析接続入門
能代清著／A5・176頁・￥680

初等微分方程式
豊田浩七著／A5・272頁・￥480

基礎数学 **微分方程式通論**
福原満洲雄・稲葉三男著／A5・￥500

応用微分方程式論
永宮健夫著／A5・274頁・￥950

常微分方程式 新版
Л. С. ポントリャーギン著
千葉克裕訳／A5・344頁・￥1000

自動制御入門のための **ラプラス変換演習**
小郷寛・佐藤真平著／A5・136頁・￥380

図形は動く
坂井豊著／B6・242頁・￥300

初等幾何学
小林幹雄著／A5・384頁・￥880

複素数の幾何学
小林幹雄著／A5・320頁・￥850

基礎数学 **統計学通論**
北川敏男・稲葉三男著／A5・244頁・￥500

新統計学
安川数太郎・内山守常著／A5・¥1200

実務家のための 数理統計学
後藤憲章著／A5・256頁・¥700

技術者のための 統計的方法
近藤良夫・舟阪渡編／A5・680頁・¥2500

ランダム変動の解析
堀川明著／A5・236頁・¥980

標準工学シリーズ 7
電子計算機 プログラミング入門
門倉敏夫著／A5・148頁・¥380

オートメーションシリーズ 5
電子計算機
磯部孝他編／A5・296頁・¥1200

電子計算機論
小南尚義著／A5・224頁・¥700

オートメーションシリーズ 6
電子計算機のプログラミング
磯部孝他編／A5・288頁・¥1200

0.01% 初等函数表
柴垣和三雄著／B5・122頁・¥580

標準工学シリーズ 14
図計算
駒井武夫著／A5・188頁・¥380

計算尺標準教本 再増補版
杉浦次郎著／A5・210頁・¥350

大学教育 図学
中根孝治著／A5・200頁・¥500

共立全書より

数学公式集
小林幹雄他編／B6・328頁・¥680

高等数学通論 I
辻正次・田中明雄著／B6・242頁・¥380

高等数学通論 II
辻正次・田中明雄著／B6・276頁・¥380

高等数学演習
辻正次・田中明雄著／B6・292頁・¥580

ギリシア数学史 I
T. L. ヒース著
平田寛他訳／B6・244頁・¥600

ギリシア数学史 II
T. L. ヒース著
平田寛他訳／B6・248頁・¥600

数と連続の哲学
白石早出雄著／B6・204頁・¥450

物理と化学のための 数学 I 改訂版
H. マージナウ・G. M. マーフィ著
佐藤次彦・国宗真訳／B6・340頁・¥700

物理と化学のための 数学 II 改訂版
H. マージナウ・G. M. マーフィ著
佐藤次彦・国宗真訳／B6・378頁・¥750

物理数学の方程式 I
C. Л. ソボレフ著
功力金二郎他訳／B6・258頁・¥650

物理数学の方程式 II
C. Л. ソボレフ著
功力金二郎他訳／B6・236頁・¥550

近代工業数学
高橋進一著／B6・232頁・¥480

技術者の 数学 I
K. S. ミラー著
佐藤常三・藤井澄二訳／B6・¥500

技術者の 数学 II
K. S. ミラー著
佐藤常三・藤井澄二訳／B6・¥450

数理論理学序説
前原昭二著／B6・234頁・¥650

行列論
遠山啓著／B6・276頁・¥550

群論
大島勝著／B6・236頁・¥600

化学者のための 群論入門
新楽和夫著／B6・198頁・¥550

束論
岩村聯著／B6・164頁・¥500

イデアル論入門
成田正雄著／B6・270頁・近刊

数論
森本清吾著／B6・210頁・¥450

整数論入門
N. M. ウィノグラードフ著
三瓶与右衛門・山中健訳／B6・¥450

無理数と極限
小松勇作著／B6・220頁・¥500

現代解析学序説
鶴見茂著／B6・196頁・¥550

積分論
河田敬義著／B6・218頁・¥450

ルベック積分
小松勇作著／B6・234頁・¥500

関数解析入門 I
M. A. ナイマルク著
功力金二郎他訳／B6・258頁・¥700

関数解析入門 II
M. A. ナイマルク著
功力金二郎他訳／B6・294頁・¥800

位相空間論
河野伊三郎著／B6・210頁・¥550

ヒルベルト空間論
吉田耕作著／B6・228頁・¥600

位相数学入門
A. H. ウォーレス著
河野伊三郎訳／B6・232頁・¥550

トポロジー入門
B. H. アーノルド著
赤摂也訳／B6・202頁・¥550

超関数論入門 I
ゲリファンド・シーロフ著
功力金二郎他訳／B6・244頁・¥550

超関数論入門 II
ゲリファンド・シーロフ著
功力金二郎他訳／B6・234頁・¥650

位相幾何学 I
P. アレクサンドロフ著
三瓶与右衛門・千葉克裕訳／B6・¥600

位相幾何学 II
P. アレクサンドロフ著
三瓶与右衛門・千葉克裕訳／B6・¥600

位相幾何学 III
P. アレクサンドロフ著
三瓶与右衛門・千葉克裕訳／B6・¥600

応用函数論
井上正雄著／B6・276頁・¥680

多変数関数論
酒井栄一著／B6・210頁・¥600

ポテンシャル論
井上正雄著／B6・280頁・¥600

新三角法
辻正次・平野智治著／B6・228頁・¥450

解析幾何学演習
小林幹雄著／B6・256頁・¥450

近代数理統計学通論
宮沢光一著／B6・320頁・¥680

数理統計学演習
宇野利雄編／B6・274頁・¥550

最大原理入門
宇野利雄・菊池豊彦著／B6・¥480

電子計算機のための 数学 I
R.W. サウスワース・S.L. デロー著
岩田倫典訳／B6・296頁・¥780

電子計算機のための 数学 II
R.W. サウスワース・S.L. デロー著
岩田倫典訳／B6・210頁・¥580

スミルノフ高等数学教程 全12巻
弥永昌吉他監修／A5・全巻揃価 10700円

市民の数学 全8冊

1. **数学の歩み**
矢野健太郎著／A5・240頁・¥650
2. **方程式と行列**
遠山啓・長妻克亘著／A5・¥650
3. **図形の性質**
穂刈四三二著／A5・228頁・¥650
4. **式とグラフと三角法**
穂刈四三二著／A5・240頁・¥680
5. **微分と積分**
矢野健太郎著／A5・178頁・¥480
6. **生活と統計**
黒田孝郎・司馬正次著／A5・¥580
7. **計算の数学**
黒田孝郎著／A5・232頁・¥650
8. **入門 新しい数学**
矢野健太郎著／A5・244頁・¥650

基礎数学講座 全21巻

1. **代数学**
奥川光太郎著／A5・202頁・¥450
2. **行列と行列式**
浅野啓三著／A5・220頁・¥450
3. **平面球面三角法**
穂刈四三二著／A5・220頁・¥450
4. **解析幾何学**
本部均著／A5・240頁・¥450
5. **綜合初等幾何学**
寺阪英孝著／A5・224頁・¥450
6. **微分積分学 I**
宇野利雄著／A5・232頁・¥450
7. **微分積分学 II**
宇野利雄著／A5・256頁・¥450
8. **微分方程式**
占部実著／A5・240頁・¥450
9. **力学**
国井修二郎著／A5・236頁・¥450
10. **確率および統計**
丸山儀四郎著／A5・210頁・¥450
11. **実用数学**
柴垣和三雄著／A5・212頁・¥450
12. **数学史・数学教授法**
小林善一他著／A5・258頁・¥450
13. **集合論**
稲垣武著／A5・234頁・¥450
14. **整数論**
稲葉栄次著／A5・226頁・¥450
15. **抽象代数学**
秋月康夫著／A5・236頁・¥450
16. **微分幾何学**
佐々木重夫著／A5・224頁・¥450
17. **函数論**
遠木幸成著／A5・224頁・¥450
18. **積分方程式・偏微分方程式**
福原満洲雄・中森寛二著／¥450
19. **実函数論および積分論**
功力金二郎著／A5・228頁・¥450
20. **近代解析**
吉田耕作著／A5・232頁・¥450
21. **位相数学**
河田敬義著／A5・224頁・¥450

数学演習講座 全15巻

1. **代数学**
奥川光太郎著／A5・192頁・¥450
2. **行列と行列式**
浅野啓三・永尾汎著／A5・¥450
3. **平面球面三角法・整数論**
穂刈四三二・稲葉三男著／¥450
4. **解析幾何学**
本部均著／A5・240頁・¥450
5. **初等幾何学**
寺阪英孝著／A5・208頁・¥450
6. **微分積分学 I**
宇野利雄他著／A5・240頁・¥450
7. **微分積分学 II**
宇野利雄他著／A5・298頁・¥450
8. **微分方程式**
占部実著／A5・194頁・¥450
9. **力学**
国井修二郎著／A5・232頁・¥450
10. **確率および統計**
丸山儀四郎他著／A5・224頁・¥450
11. **集合論・抽象代数学**
稲垣武他著／A5・204頁・¥450
12. **微分幾何学**
佐々木重夫著／A5・220頁・¥450
13. **函数論**
遠木幸成著／A5・210頁・¥450
14. **偏微分方程式**
福原満洲雄・中森寛二著／¥450
15. **積分論・位相解析**
功力金二郎他著／A5・236頁・¥450

現代数学講座 全25巻より

1. **数学基礎論**
竹内外史著／A5・114頁・¥400
2. **近代代数学**
秋月康夫・永田雅宜著／A5・¥650
4. **代数的整数論**
河田敬義著／A5・188頁・¥650

7. フーリエ解析
洲之内源一郎著／A5・130頁・¥450

8. 幾何学的函数論
遠木幸成著／A5・96頁・¥400

10. 多変数函数論
一松信著／A5・122頁・¥400

11. 微分方程式論
福原満洲雄・佐藤徳意著／¥1000

13. 超函数論
吉田耕作著／A5・178頁・¥700

15. リー環論
松島与三著／A5・196頁・¥800

16. 代数幾何学
中井喜和・永田雅宜著／A5・¥700

17. 射影幾何学
秋月康夫・滝沢精二著／A5・¥550

19. 接続の幾何学
大槻富之助著／A5・182頁・¥600

20. 積分幾何学
栗田稔著／A5・120頁・¥400

21. 位相幾何学
小松醇郎他著／A5・¥850

22. 確率論
丸山儀四郎著／A5・120頁・¥500

23. 数理統計学 改訂版
工藤弘吉著／A5・136頁・¥500

24-A 非線形問題 自励振動論
占部実著／A5・146頁・¥500

25. 特殊函数論
柴垣和三雄著／A5・176頁・¥600

新しい数学へのアプローチ 全巻15より

1. 集合
矢野健太郎著／A5・144頁・¥550

2. 論理
中谷太郎著／A5・200頁・¥580

3. ベクトル
穂刈四三二著／A5・232頁・¥650

5. 新しい代数
本部均著／A5・208頁・近刊

6. トポロジー
中村勝彦著／A5・186頁・¥580

7. 複素数
小林善一著／A5・170頁・¥500

8. いろいろな幾何
中沢貞治著／A5・186頁・¥580

9. いろいろな曲線
栗田稔著／A5・176頁・¥580

10. 微積分の基礎
小西勇雄著／A5・210頁・¥600

11. 微分方程式
石原繁著／A5・160頁・¥550

12. 確率
渡部隆一著／A5・184頁・¥580

13. 統計 増訂版
岡田泰栄著／A5・210頁・¥600

14. 線形計画法
千住鎮雄著／A5・160頁・¥550

15. 電子計算機
刀根薫・恒川純吉著／A5・¥650

共立数学講座 全25巻／別巻2より

1. 解析学概論 I
山崎圭次郎著／A5・304頁・¥680

3. 線形数学 I
入江昭二著／A5・178頁・¥500

4. 線形数学 II
入江昭二著／A5・270頁・近刊

7. 初等代数学
成田正雄著／A5・256頁・¥680

16. 線形位相空間と一般関数
山中健著／A5・206頁・¥600

18. 群とその表現
服部昭著／A5・228頁・¥680

19. 多様体
村上信吾著／A5・240頁・近刊

23. 最適問題
杉山昌平著／A5・208頁・¥600

共立構座 現代の数学 全35巻より

6. アーベル群・代数群
本田欣哉・永田雅宜著／A5・¥1000

9. 代数幾何学入門
中野茂男著／A5・220頁・¥1000

19. ノルム環
和田淳蔵著／A5・232頁・¥1200

21. ポテンシャル論
二宮信幸著／A5・220頁・近刊

28. 数値解析の基礎
山口昌哉・野木達夫著／A5・近刊

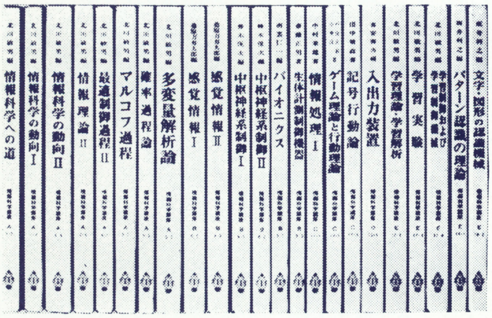
21世紀の科学・技術のビジョン
探求し，情報科学から情報素子

現代科学のパイオニア

情報科学講座

全65巻

（編集委員）
大泉充郎
勝木保次
北川敏男
喜安善市
栗原俊彦
桑原万寿太郎
坂井利之
高田昇平
次田皓
南雲仁一
中村幸雄
和田弘



■既刊

A・1・1 **情報科学への道**
北川敏男編／156頁・¥600

A・1・2 **情報科学の動向Ⅰ**
北川敏男編／190頁・¥700

A・1・3 **情報科学の動向Ⅱ**
北川敏男編／272頁・¥950

A・2・5 **情報理論Ⅱ**
北川敏男編／154頁・¥600

A・4・2 **最適制御過程Ⅱ**
北川敏男編／160頁・¥700

A・5・1 **マルコフ過程**
北川敏男編／202頁・¥800

A・5・2 **確率過程論**
北川敏男編／120頁・¥500

A・5・3 **多変量解析論**
北川敏夫編／254頁・¥950

B・6・1 **感覚情報Ⅰ**
桑原万寿太郎編／218頁・¥800

B・6・2 **感覚情報Ⅱ**
桑原万寿太郎編／200頁・¥750

B・8・1 **中枢神経系制御Ⅰ**
勝木保次編／168頁・¥650

B・8・2 **中枢神経系制御Ⅱ**
勝木保次編／138頁・¥600

B・9・1 **バイオニクス**
南雲仁一編／122頁・¥500

B・9・3 **生体計測制御機器**
斉藤正男著／96頁・¥400

C・11・1 **情報処理Ⅰ**
中村幸雄著／128頁・¥600

C・12・1 **ゲーム理論と行動理論**
戸田正直・中原淳一著／118頁・¥550

C・12・3 **記号行動論**
田中靖政著／138頁・¥600

D・13・1 **入出力装置**
橋本南海男著／200頁・¥800

D・13・2 **論理素子**
喜安善市編／160頁・¥650

共立出版

〔各巻〕A5判　上製函入

を求める若い科学者・技術者のために，情報革命の姿を
および情報処理へつながる道を明らかにする未来の書…

D・13・3 **記憶装置Ⅰ**
喜安善市編／216頁・¥850

E・17・1 **学習理論・学習解析**
北川敏男編／192頁・¥800

E・17・3 **学習実験**
北川敏男編／124頁・¥500

E・17・4 **学習制御および学習制御機械**
北川敏男編／202頁・¥800

E・19・1 **パターン認識の理論**
坂井利之編／114頁・¥500

E・19・2 **文字・図形の認識機械**
坂井利之編／184頁・¥900

■続刊

A 共通基礎理論

A・1・4 情報科学の将来
A・2・1 論理数学Ⅰ
A・2・2 論理数学Ⅱ
A・2・3 論理数学Ⅲ
A・2・4 情報理論Ⅰ
A・2・6 情報理論Ⅲ
A・3・1 計画理論概説
A・3・2 数理計画法
A・3・3 決定理論
A・3・4 組織論
A・4・1 最適制御過程Ⅰ
A・4・3 制御原理
A・4・4 統計的制御過程
A・4・5 適応制御過程
A・5・4 推測過程論

B 生体情報

B・7・1 遺伝情報Ⅰ
B・7・2 遺伝情報Ⅱ
B・8・3 中枢神経系制御Ⅲ
B・9・2 人間−機械系

C 言語および行動

C・10・1 言語理論
C・10・2 言語の機械処理
C・10・3 計算機用言語
C・11・2 情報処理Ⅱ
C・11・3 情報処理Ⅲ
C・12・2 情報の伝播
C・12・4 グループダイナミックス

D 情報システム

D・13・4 記憶装置Ⅱ
D・13・5 情報素子の小型実装法
D・13・6 新しい情報素子
D・13・7 オプトエレクトロニクス
D・14・1 計算体系Ⅰ
D・14・2 計算体系Ⅱ
D・15・1 計算機の応用Ⅰ
D・15・2 計算機の応用Ⅱ
D・16・1 実時間システムⅠ
D・16・2 実時間システムⅡ

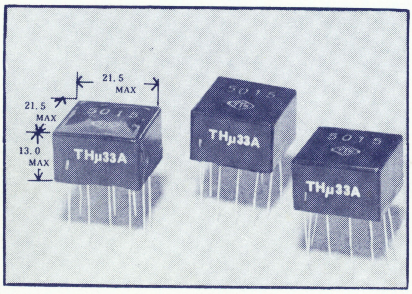
E 情報系モデル

E・17・2 プログラム学習
E・18・1 自己組織化モデルⅠ
E・18・2 自己組織化モデルⅡ
E・19・3 音声の認識機械

信頼と技術を誇る

東洋通信機の安定化された小型電源

THμ33Aシリーズはエポキシモールドされた厚膜混成集積回路で、半導体集積回路を使用したディジタル機器の電源回路に最適であります。優れた回路技術と製造技術により電圧設定確度が±0.5%、0～70°Cでの温度変動が±1%など特性が優れております。出力ダーリントントランジスタや、誤差増幅回路、基準電圧素子などはTO－5の金属容器に封入されており、堅牢な厚膜抵抗基板の使用とともに通信工業用として充分な信頼性を有します。又全くこれと極性を反転した負電源回路THμ33Bやスレーブ専用のもの、より高い出力電圧のTHμ34シリーズも製作いたします。



特　長

- 出力電圧の設定確度が高い。
- 温度特性が優れている。
- 信頼度が高い。
- 過電流保護（限流）回路が内蔵されている。
- 外部に附加する部品が少ない。
- 種々の使用法が可能で幅広いシステムへの適用性を有する。

THμ33Aシリーズ定格表

電気的特性

MODEL		3015	4715	5015	5215	5515
出力電圧		3.0V±15mV	4.7V±20mV	5.0V±25mV	5.2V±25mV	5.5V±30mV
出力電流		0～200mA				
消費電力		0.5W MAX				
入力電圧範囲		4～20V	6～20V	6～20V	6.2～20V	6.5～20V
負荷変動(0～0.1A)		5mV MAX				
電源変動(無出力時)	入力1	5mV MAX	5mV MAX	7mV MAX	10mV MAX	10mV MAX
	入力2	5mV MAX				
温度特性※0－70°C		±30mV MAX	±47mV MAX	±50mV MAX	±52mV MAX	±55mV MAX
消費電流		4mA MAX				

※更に広い温度範囲、狭い変動範囲のものも製作します。

TTK 東洋通信機株式會社

本社川崎工場	神奈川県川崎市塚越3丁目484番地	電話川崎(044) 52-4111(大代表)
相模事業所 水晶事業部 相模工場	神奈川県高座郡寒川町小谷753番地	テレックス 3842-185 電話寒川(0467) 75-1131(大代表) テレックス 3862-133
大阪支社	大阪市東区淡路町5丁目2番地（長谷川第1ビル）	電話大阪(06) 203-7485(代表)
東京営業所	東京都港区芝虎ノ門15番地（虎ノ門ビル）	電話東京(03) 503-3441(代表)
福岡営業所	福岡市天神2丁目12番1号（天神ビル）	電話福岡(092) 75-6031・6416

通常払込料金加入者負担

払込通知票

各票の※印欄は払込人において記載して下さい。

文字は正確明りように、数字はアラビア数字を使ってお書き下さい。

- 口座番号：東京 57035（十・万・千・百・十・番）
- 加入者名：共立出版株式会社
- 金額：※（億・千・百・十・万・千・百・十・円）
- 払込人住所氏名：※
- 備考
- 受付局日附印

（郵政省）

通常払込料金加入者負担

払込票

記載事項を訂正した場合は、その箇所に証印して下さい。

各票の記載事項に間違のないことをお確め下さい。

- 口座番号：東京 57035（十・万・千・百・十・番）
- 加入者名：共立出版株式会社
- 金額：※（億・千・百・十・万・千・百・十・円）
- 払込人住所氏名：※
- 料金：払込 ／ 特殊 円
- 備考
- 受付局日附印
- 局番号印

（郵政省）

ご注文の方法について

直接小社ご注文ご希望の方は書籍定価に下記相当額の送料をご加算の上，ご送金下さい。なお直送の場合はお手許に届くまでに10日から14日位の日数を要します。その点お含み下さるようお願い致します。

書籍小包郵便料金

重量／あて地	250g まで	500g まで	1 kg まで	1.5kg まで	2 kg まで
全国均一	50円	70円	90円	110円	120円

ただし1冊送本の場合に限ります。なお速達ご希望の方は上記の金額に80円をご加算下さい。

コンピュータ・サイエンス誌

bit

年間購読料 3120円（送料小社負担）

申込　　　月号より

なお創刊号（3月号）より在庫揃っております

月号	定価	送料	冊数	備考
3月号	260円	50円	冊	
4月号	260円	50円	冊	
5月号	260円	50円	冊	
6月号	260円	18円	冊	
7月号	260円	18円	冊	

通信欄

JAC 電子計算機

JAC-110形

性能に比べて廉価です

●記憶容量が多く、使いやすく、ソフトウエアサービスが行きとどいていて、しかも価格は会計機並という理想的な電子計算機です。

●そのわけは、自社製のトランジスタ、ダイオード、ＩＣと記憶装置を使っており、五十年にわたるエレクトロニクスのトップメーカの設計だからです。

適用業務

●販売管理・資材管理・商品管理・財務管理・原価計算・給与計算・会計業務●病院業務●税金業務・保険業務・料金調停業務●農協・漁協各種精算業務●各種科学技術計算

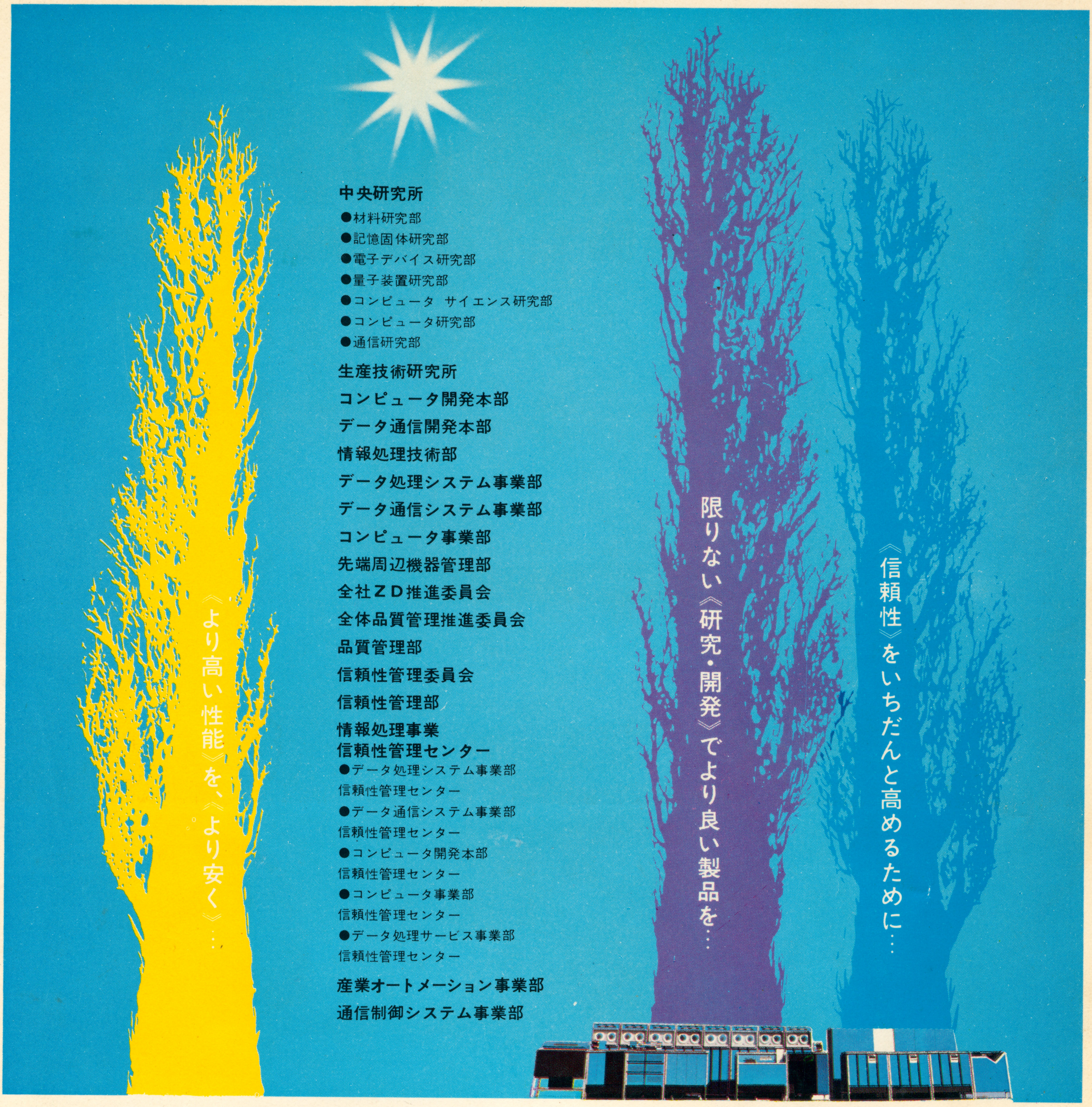
世界の電波をつくる

JRC 日本無線

東京都港区芝桜川町25　第5森ビル
電話 東京（591）（大代）3451
大阪市北区堂島中1の23　堂島中町ビル
電話 大阪（344）（大代）1631
福岡市渡辺通り4の9の18号　福酒ビル
電話 福岡（76）（代）2636

札幌・仙台・八戸・清水・名古屋・神戸
舞鶴・呉・広島・長崎・熊本・鹿児島

bit 7月号
昭和44年7月1日発行／毎月1日発行／通巻第5号／昭和44年5月2日第3種郵便物認可／昭和44年1月18日国鉄東局特別承認雑誌第3065号
定価260円



NECコンピュータは21世紀を目ざして限りない成長をつづけています

●研究・開発 ●コストパフォーマンスの向上 ●信頼性確保のための万全の体制をご覧ください

情報は巨大化する。複雑化する。コンピュータは、私たちの暮しやビジネスの中で、現在から未来へ、飛躍的に必要の度合いを高めていきます。こんな中で、NEC日本電気は万全の体制をととのえ、5年・10年後、さらに21世紀へと遠い将来まで見通して、最先端をいく研究・開発に取り組んでおります。明日の情報化時代にも、安心して使えるコンピュータを、いち早くお届けしているのです。

NEAC〈ニアック〉はNECコンピュータの製品名です
●NECコンピュータは日本電子計算機株式会社を通じてレンタル制を実施しております。

NEC コンピュータ
NEAC
日本電気
お問合せは：データ処理システム事業部営業企画部へ
東京都港区芝五丁目33番8号（田町ビル）
TEL（03）452-1111（大代表）〒108
本社 東京都港区芝五丁目7番15号
TEL（03）452-1111（大代表）〒108
支社・支店・営業所 大阪・札幌・仙台・郡山・鹿島
新潟・富山・金沢・千葉・大宮・横浜・静岡・浜松
名古屋・京都・姫路・岡山・広島・徳山・高松・松山
北九州・福岡・大分・鹿児島

雑誌コード 7607

bit

コンピュータ・サイエンス誌

Vol. 1 No. 5 通巻第5号